IEEE Standards for Local and Metropolitan Area Networks: Integrated Services (IS) LAN Interface at the Medium Access Control (MAC) and Physical (PHY) Layers

Sponsor

Technical Committee on Computer Communications of the IEEE Computer Society

Approved September 22, 1994

IEEE Standards Board

Abstract: A unified access method that offers integrated services (IS) to the desktop for a variety of publicly and privately administered backbone networks (e.g., ANSI FDDI, IEEE 802.x, and ISDN) is defined. In addition, the interface at the MAC sublayer and the PHY Layer is specified.
Keywords: access unit (AU), data link layer, hybrid multiplexer (HMUX), integrated services digital network (ISDN), integrated services terminal equipment (ISTE), layer management entity, local area network (LAN), logical link control, managed object, management information base (MIB), medium access control (MAC) sublayer, metropolitan area network (MAN), physical (PHY) layer, physical medium dependent, physical signalling, private switching network, protocol data unit (PDU), service access point, time division multiplexer (TDM)

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

ISBN 1-55937-456-X

IEEE Standards documents are developed within the Technical Committees of the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason IEEE and the members of its technical committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

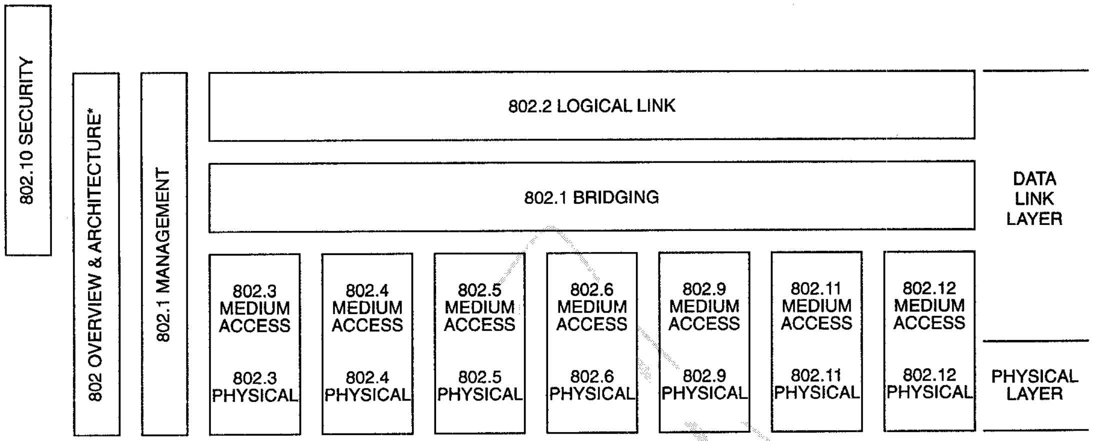
Secretary, IEEE Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

IEEE standards documents may involve the use of patented technology. Their approval by the Institute of Electrical and Electronics Engineers does not mean that using such technology for the purpose of conforming to such standards is authorized by the patent owner. It is the obligation of the user of such technology to obtain all necessary permissions.

Introduction

[This introduction is not a part of IEEE Std 802.9-1994, IEEE Standards for Local and Metropolitan Area Networks: Integrated Services (IS) LAN Interface at the Medium Access Control (MAC) and Physical (PHY) Layers.]

This standard is part of a family of standards for local and metropolitan area networks. The relationship of the members of the family to each other is shown below. (The numbers in the figure refer to IEEE standard numbers.)



* Formerly IEEE Std 802.1A.

This family of standards deals with the Physical and Data Link Layers as defined by the International Organization for Standardization (ISO) Open Systems Interconnection Basic Reference Model (ISO 7498 : 1984). The access standards define several types of medium access technologies and associated physical media, each appropriate for particular applications or system objectives. Other types are under investigation. The IEEE 802.9 ISLAN Interface Standard specifies both isochronous and packet mode services, and provides access to ISDN and other backbone networks for service provisioning.

The standards defining these technologies are as follows:

- IEEE Std 802[a]: Overview and Architecture. This standard provides an overview to the family of IEEE 802 Standards. This document forms part of the 802.1 scope of work.

- ISO/IEC DIS 15802-2 [IEEE Std 802.1B and 802.1k]: LAN/MAN Management. Defines an Open Systems Interconnection (OSI) management-compatible architecture, and services and protocol elements for use in a LAN/MAN environment for performing remote management.

- ISO/IEC 10038 [ANSI/IEEE Std 802.1D]: MAC Bridging. Specifies an architecture and protocol for the interconnection of IEEE 802 LANs below the MAC service boundary.

[a]The 802 Architecture and Overview Specification, originally known as IEEE Std 802.1A, has been renumbered as IEEE Std 802. This has been done to accommodate recognition of the base standard in a family of standards. References to IEEE Std 802.1A should be considered as references to IEEE Std 802.

• ISO/IEC 15802-4 [ANSI/IEEE Std 802.1E]:	System Load Protocol. Specifies a set of services and protocol for those aspects of management concerned with the loading of systems on IEEE 802 LANs.
• ISO/IEC 8802-2 [ANSI/IEEE Std 802.2]:	Logical Link Control
• ISO/IEC 8802-3 [ANSI/IEEE Std 802.3]:	CSMA/CD Access Method and Physical Layer Specifications
• ISO/IEC 8802-4 [ANSI/IEEE Std 802.4]:	Token Bus Access Method and Physical Layer Specifications
• ISO/IEC 8802-5 [ANSI/IEEE Std 802.5]:	Token Ring Access Method and Physical Layer Specifications
• ISO/IEC 8802-6 [ANSI/IEEE Std 802.6]:	Distributed Queue Dual Bus Access Method and Physical Layer Specifications
• IEEE Std 802.9:	Integrated Services (IS) LAN Interface at the Medium Access Control (MAC) and Physical (PHY) Layers
• IEEE Std 802.10:	Interoperable LAN/MAN Security, *Currently approved:* Secure Data Exchange (SDE)

In addition to the family of standards, the following is a recommended practice for a common technology:

• IEEE Std 802.7:	IEEE Recommended Practice for Broadband Local Area Networks

The following additional working groups have authorized standards projects under development:

• IEEE 802.11:	Wireless LAN Medium Access Control (MAC) and Physical Layer Specifications
• IEEE 802.12:	Demand Priority Access Method/Physical Layer Specifications
• IEEE 802.14:	Standard Protocol for Cable-TV Based Broadband Communication Network

Conformance test methodology

An additional standards series, identified by the number 1802, has been established to identify the conformance test methodology documents for the 802 family of standards. Thus the conformance test documents for 802.3 are numbered 1802.3, the conformance test documents for 802.5 will be 1802.5, and so on. Similarly, ISO will use 18802 to number conformance test standards for 8802 standards.

The ongoing work of the IEEE 802 committee has resulted in standards for data communications in a local area network (LAN) environment. As office workstations have proliferated, however, the demand for LANs has substantially increased. This has led to the inevitable diversification in market requirements.

Since the typical office worker requires access to both data and voice services, among others, at the desktop, there has been a growing trend toward integrated services (IS), namely voice, data, and video. Due to the increasing need for facsimile, image transfer, and video services, these services are included in the general category of the integration of voice and data services. Such integration offers potential economies to the business customer in terms of reduced components (one port per station instead of two or more), and in simpler management and maintenance (one network instead of two or more).

The provision of voice service is generally effected using unshielded twisted-pair wire. Not only is this medium widespread in typical office environments, but it is also inexpensive and easy to install and maintain. In the vast majority of installations, there is spare capacity, and in these cases, the use of such a medium is essentially free since the need for rewiring is significantly reduced. Moreover, with existing technology, it is possible to provide medium- to high-performance data service over the unshielded twisted-pair wire. This places special emphasis on the use of the unshielded twisted-pair wire to provide IS services. This standard extends the scope and capability of existing twisted-pair wiring and thus reduces the incentive for overlay wiring systems.

With respect to the provision of integrated services, there has been ongoing work in other standards bodies, notably, the International Telecommunication Union–Telecommunication Standardization Sector (ITU-T) on the provision of such services through integrated services digital networks (ISDNs). While the principal focus has been the provision of such services using public networks, efforts are under way [notably, in the European Computer Manufacturers Association (ECMA)] to extend such services to customer premises networks.

This standard defines a unified access method that offers global integrated services to the desktop for a variety of publicly and privately administered backbone networks (e.g., ANSI FDDI, IEEE 802.x[b], and ISDN). This standard will enable ISTEs to be attached to IEEE 802.9 LANs and will allow them to communicate with other IS stations as well as data-only stations, voice-only stations, and premises-based networks offering ISDN services. In addition, it specifies the use of unshielded telephone twisted pair as the primary medium of distribution.

The use of terminal adaptor (TA) devices will permit the direct coupling of native mode terminal devices such as data-only modules, voice modules, and ISDN basic rate terminals to the IEEE 802.9 interface. This standard has been designed to accommodate the adaptation of ISDN basic rate station devices and IEEE 802.x station devices to the IEEE 802.9 interface. In summary, this standard represents the integration of IEEE 802 services and ISDN services.

[b]IEEE 802.x refers to the entire family of IEEE 802 standards.

Participants

When the IEEE Project 802.9 Working Group approved IEEE Std 802.9-1994, it had the following membership:

Dhadesugoor R. Vaman, *Chair*
Ronald Kemper Sr., *Vice Chair*
Wayne A. Zakowski, *Editor*
George Westwick, *Secretary*

Sven Olof Akerlund
Dan Ater
Kwame Boakye†
John Boal‡
Richard Brand
Dale Buchholz
Allan Cobb*
John Conville
Felix Diaz
Earl Emerson
Staffan Fredricsson
Ingrid Fromm
Stephen Hall
Tohru Kazawa
Kansuke Kurayanagi
William Lee
Kiyoshi Maruyama
Takanori Miyamoto
John Moughton
Tony Parker
Keith Patterson*
Sanjay Popli
Floyd Ross
Everett Rigsbee
Chandos Rypinski‡
Hiroshi Shimizu
Toshiro Suzuki
Kenta Takumi
Norman Tester
Ian Thomas
Kazuo Unemoto
Surya Varanasi
Kojiro Watanabe

* Past Chair
† Past Editor
‡ Past Secretary

The following persons were on the balloting committee:

William B. Adams
Albert Bernhard
Robert M. Amy
Yong Myung Baeg
Asa O. Bishop
Michael J. Caldwell
George Carson
Basilio Chen
Robert Crowder
Robert A. Donnan
Sourav Kumar Dutta
Judith A. Edwards
Philip H. Enslow
Changxin Fan
John W. Fendrich
Harvey A. Freeman
Ingrid Fromm
Robert Gagliano
D. G. Gan
Isaac Ghansah
Julio Gonzalez-Sanz
G. L. Hollander
Paul L. Hutton
Richard J. Iliff
Raj Jain
Anura P. Janasumana
Gary C. Kessler
Youngbum Kim
Demosthenes Kostas
Randolph S. Little
Ming T. Liu
Donald C. Loughry
Nam C. Low
Gottfried W. R. Luderer
Peter Martini
Richard McBride
William C. McDonald
A. E. Methiwalla
Ann Miller
David S. Millman
John E. Montague
Kinji Mori
David J. Morris
D. O'Mahony
Charles Oestereicher
Thomas E. Phillips
Art J. Pina
Yang Qianli
Fernando Ramos
John P. Riganati
Edouard Y. Rocher
Robert Rosenthal
Floyd E. Ross
Rainer A. Rueppel
Christoph Ruland
Norman Schneidewind
Adarshpal S. Sethi
Donald A. Sheppard
Leo Sintonen
Saragur M. Srinidhi
Joseph Stevenson
Clay Stocklin
Fred J. Strauss
E. D. Sykas
Ahmed N. Tantawy
Geoffrey O. Thompson
Robert Tripi
Mark-Rene Uchida
Dhadesugoor R. Vaman
Raymond Van Houtte
Punj Vikram
James T. Vorhies
Garry M. Vornbrock
Alan J. Weissberger
Raymond Wenig
A. Jameson West
Oren Yuen

Contents

CHAPTER		PAGE

IEEE Standards for Local and Metropolitan Area Networks: Integrated Services (IS) LAN Interface at the Medium Access Control (MAC) and Physical (PHY) Layers

1. Overview

1.1 Scope and purpose

The scope of this standard is to

- Develop an ISLAN interface at the medium access control (MAC) sublayer and the Physical (PHY) Layer compatible with IEEE 802.x[1] and ISDN standards and architectures
- Develop an ISLAN interface that operates independently from the backbone network
- Focus upon unshielded twisted-pair wiring as the primary distribution medium
- Enable implementation of IS terminal equipment (ISTE) that accesses IEEE 802 LAN and ISDN services through a common interface

The body of this standard

- Defines the service provided by the MAC sublayer to the IEEE 802 Logical Link Control (LLC) sublayer and management, and describes the services provided by the PHY Layer to the MAC sublayer and management in terms of service primitives and associated parameters
- Describes the services provided by the Physical Layer (PHY MUX) to support a basic rate interface (BRI) ISDN in terms of service primitives and associated parameters
- Describes the services provided by the PHY MUX to the isochronous channels
- Specifies the MAC functions that allow ISTEs access to one another and to LANs providing IEEE 802 services and/or ISDN services
- Specifies the frame format for the MAC frame
- Defines the MAC protocol
- Specifies the channel structure and frame format of the time division multiplexed (TDM) frame

[1]IEEE 802.x refers to the entire family of IEEE 802 standards.

— Specifies the PHY Layer functions over unshielded telephone twisted-pair (UTTP) cable
— Specifies the characteristics of the UTTP attachment of the station to the access unit (AU) including the specification of the medium interface connector
— Specifies the definition of MAC and PHY managed objects (MOs)
— Describes recommended ISDN signalling and management methods to coordinate the multiple channels operated between ISTEs and the AU

The normative annexes provide

— The Protocol Implementation Conformance Statement (PICS) proforma
— The Managed Object Conformance Statement (MOCS)
— The Guidelines for the Definition of Managed Objects (GDMO) specifications
— Supplemental recommendations on the use of CCITT Q.93x signalling procedures

The informative annexes provide

— Reference models of common configurations as guidelines for implementation
— A sequence and description language (SDL) description of the information flow across protocol layers
— Description of an optional remote secure control (RSC) procedure to invoke secure communication device (SCD) operations
— Recommendations for a common, consistent synthesis of ISDN and IEEE 802 that conforms with the addressing of ISTE devices and services
— Description of a mechanism to support the transport of a broadband ISDN conformant "cell" bearer service

1.2 Notation

1.2.1 Service specification method and notation

This subclause describes the method of specification of the services required of the MAC sublayer by the LLC sublayer as well as of the PHY Layer by the MAC sublayer of the P channel, the Data Link Layer of the D channel, and the services provided on the B and C channels.

In general, the services of a layer (or sublayer) are the capabilities that it offers to a user in the next higher layer (or sublayer). In order to provide its services, a layer builds its functions on the services that it requires from the next lower layer. Figure 1-1 illustrates this notion of service hierarchy and shows the relationship of the two corresponding N Layer users.

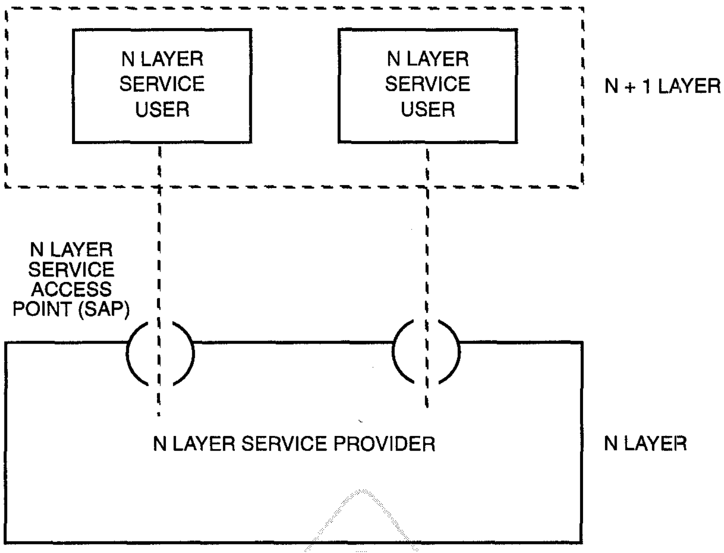


Figure 1-1—Layer service model

This information flow is modeled by discrete, instantaneous events that characterize the provision of a service. Each event consists of passing a service primitive from one layer to the other through an N Layer service access point (SAP) associated with an N+1 Layer service user. These SAPs are shown in figure 1-1. Service primitives convey the information required to provide 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 layer service is independent of any particular interface implementation, and is not subject to conformance testing requirements.

In order to comply with the service message flow signals described in the CCITT Recommendations on ISDN, it is necessary to consider all four primitive flow types.

a) *Request primitive.* This primitive is passed from the N+1 Layer service user to the N Layer (or sublayer) to request that a service be initiated.

b) *Indication primitive.* This primitive is passed from the N Layer (or sublayer) to the N+1 Layer service user to indicate an internal N Layer event that is significant to the N+1 Layer service user. This may be logically related to a remote service request, or may be caused by an event internal to the N Layer.

c) *Response primitive.* This primitive is passed from the N+1 Layer service user to the N Layer (or sublayer) to complete a procedure previously invoked by an indication primitive.

d) *Confirm primitive.* This primitive is passed from the N Layer (or sublayer) to the N+1 Layer service user to convey the results of one or more associated previous service requests.

Figure 1-2 shows the service primitives and N+1 Layer peer protocol entities associated with the two corresponding N+1 Layer service users. Services are specified by describing the service primitives and parameters that characterize each service. A service may have one or more related primitives that constitute the activity that is related to the particular service. Each service primitive may have zero or more parameters that convey the information required to provide the service.

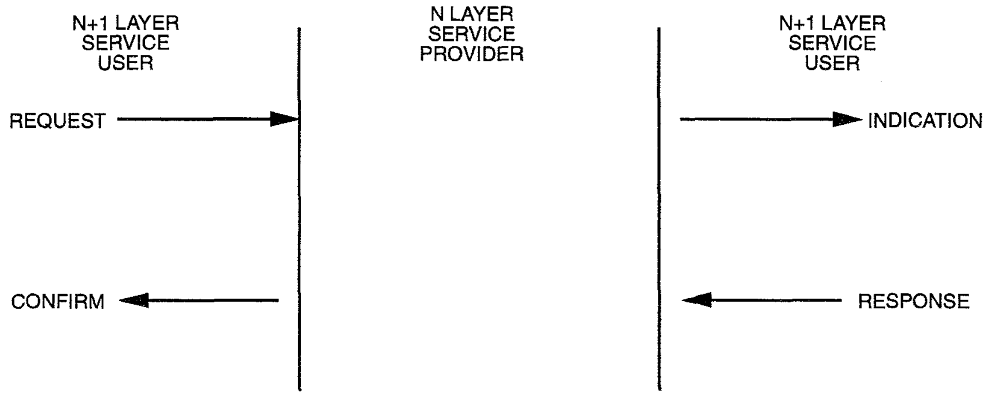


Figure 1-2—Service primitives

Figure 1-3 illustrates the end-to-end relationship of the service primitives and the N Layer peer protocol entities across a communication network.

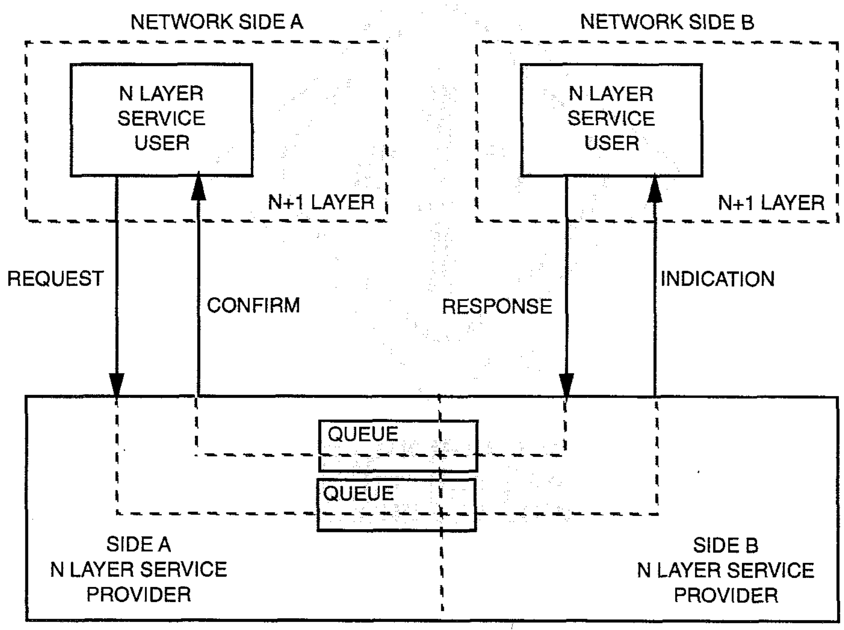


Figure 1-3—End-to-end relationship of service primitives

1.2.2 Timing relationship between service primitives

Figure 1-4 a) illustrates the flow from the service user to the service provider for a request. Figure 1-4 b) illustrates the notification to the service user of an event that has occurred in the service provider. Figure 1-4 c) illustrates the event wherein two separate service users are simultaneously making a service request at each end of the peer communication. Figure 1-4 d) illustrates the event wherein both ends of a communication link have the service provider sending up an indication of an event occurrence. Figure 1-4 e) illustrates the common situation wherein a "request" from a service user at one side of a network is sent in the form of a message action across the network to the peer side. The service provider at the other end of the network will report the incoming protocol packet as an "indication." Figure 1-4 f) depicts a communication flow in which all four types of primitives are used as part of a communication between two ends of the network.

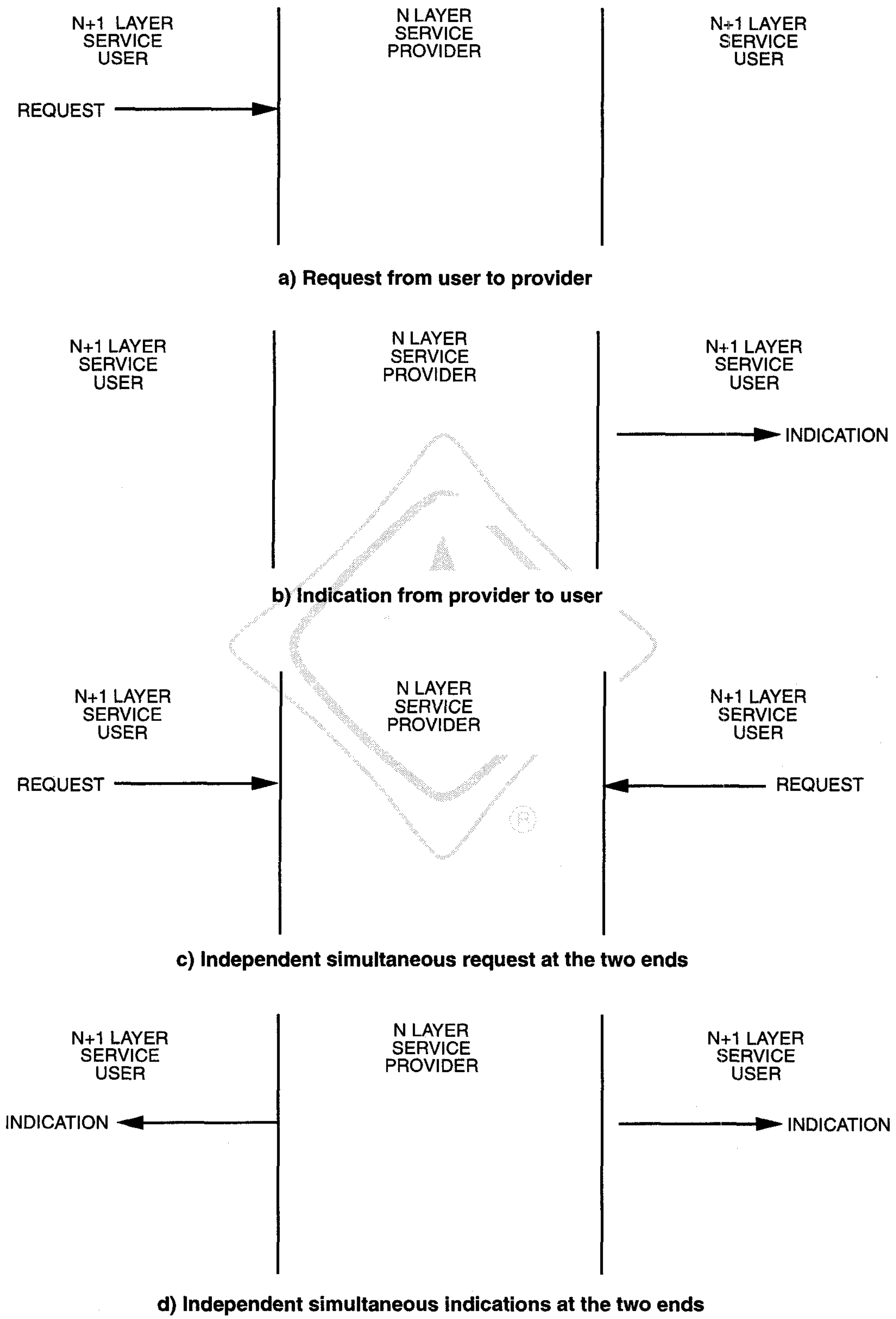
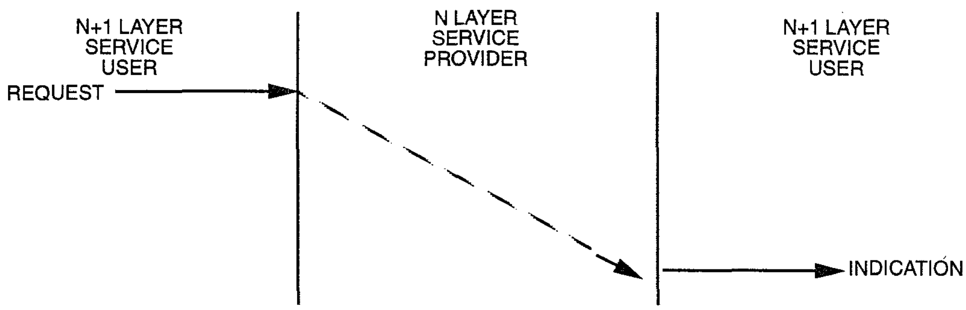
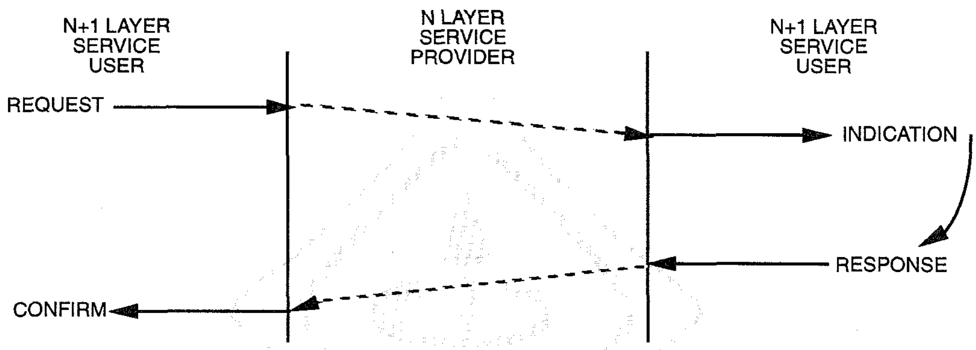


Figure 1-4—Timing relationship between service primitives



e) Request leading to an indication at the peer entity



f) Cycle of interactions between peer entities

Figure 1-4—Timing relationship between service primitives

1.3 Service model

In the general N Layer service model, the N+1 Layer service user communicates with the N Layer service provider via individual request and indication primitives. In theory, this model requires that the transmit and receive processes in the N+1 Layer be synchronized to the N Layer in a serial fashion.

In the OSI model there is a presumption of sending and receiving application layer independence. This is inclusive of asynchronous service operation and hence there shall be at least one queue between communicating peer operations. However, it does not make any particular layer or layer boundary responsible for effecting the required queue.

Furthermore, the OSI model provides no means of describing the conveyance of timing information that is necessarily associated with the data request and indication primitives of an isochronous service.

It is now widely recognized that the description of ISDN "out-of-band" signalling services is best modeled by a coordination function that associates lower level control plane and user plane stacks.

Juniper Ex 1032 - 16
Juniper v Chrimar
IPR2016-01399

1.3.1 Isochronous services

The IEEE 802.9 ISLAN employs a TDM bearer mechanism within the PHY Layer. This permits the support and delivery of a plurality of transparent separate isochronous service channels, the provision of an octet alignment signal for these channels, and a facility to provide (optionally) and accept precise timing signals.

It is the provision of the timing signal that principally distinguishes the isochronous services from the asynchronous packet services that IEEE 802.9 provides. In 9.2, the PHY interface service specification includes in addition to received and transmitted data signals, a clock indication (and optionally clock request) as well as framing (TDM) indication and request signals. It is these additional signals that permit the IEEE 802.9 PHY Layer to support the conveyance of isochronous information.

In delivering an ISDN service, it is essential that the information rate at an ISTE exactly match that of the ISDN network to which the host IEEE 802.9 AU is attached. Thus, in the case of the AU it is a requirement that the TDM bearer clock rate recovered from the ISDN network can be input to the IEEE PHY for conveyance to a supported ISTE/TA. For this reason an IEEE 802.9 AU PHY transmit clock is an input signal while at the IEEE 802.9 ISTE/TA the signal is an output indicating the time at which sequential PHY data request signals will be serviced.

The details of the abstract PHY service specification can be found in 7.1.1.5.

1.3.2 Service queues between layers and layer management

The service channel provided by the IEEE 802.9 PHY Layer for support of the OSI Data Link Layer functions of the IEEE 802.9 is not distinct from any other PHY service channel; it too is isochronous. However, it is neither possible nor required that the MAC sublayer should provide an isochronous service to the IEEE 802 LLC sublayer. It is in the IEEE 802.9 MAC sublayer that the timing information conveyed by the PHY is discarded. In practice, it is conventional to implement within the MAC sublayer a first-in first-out (FIFO) queue both for the send and receive paths. This advantageously allows decoupling of the PHY, MAC, and LLC processes. It is not, however, a requirement.

However, the interface between station management processes and both the IEEE 802.9 PHY and MAC Layers must be asynchronous since the station management process is an Application Layer entity. It follows, therefore, that there must be a queue between the PHY Layer and its layer management and between the MAC Layer and its layer management.

To allow for the adjoining layers to operate more asynchronously, the N+1 Layer's transmit request may be stored in a transmit work queue wherein the N Layer may choose to serve this request at a later instant in time. Likewise, the N Layer may store its receive messages (in correct time order) in a receive queue. This enables the N+1 Layer to respond to the incoming message at a later instant in time.

Figure 1-5 illustrates this asynchronous linkage between layers. The queues are shown with dashed lines to depict the dynamic length of the given queue. It should be noted that at a given instant in time, the length of the queue may be zero and there may be no measurable delay in the passing of request/indication signals. The general model allows for queues between any two adjoining layers. Since the N Layer management entity may wish to send information to its peer layer management entity via the N Layer, figure 1-5 also shows queues between the N Layer and its associated layer management entity. The management of these queues is the responsibility of layer management. This management/control of the queues is shown with the dotted "control" signals.

The transmit and receive queues that are illustrated between the LME entity and the N Layer are for peer-to-peer LME exchange.

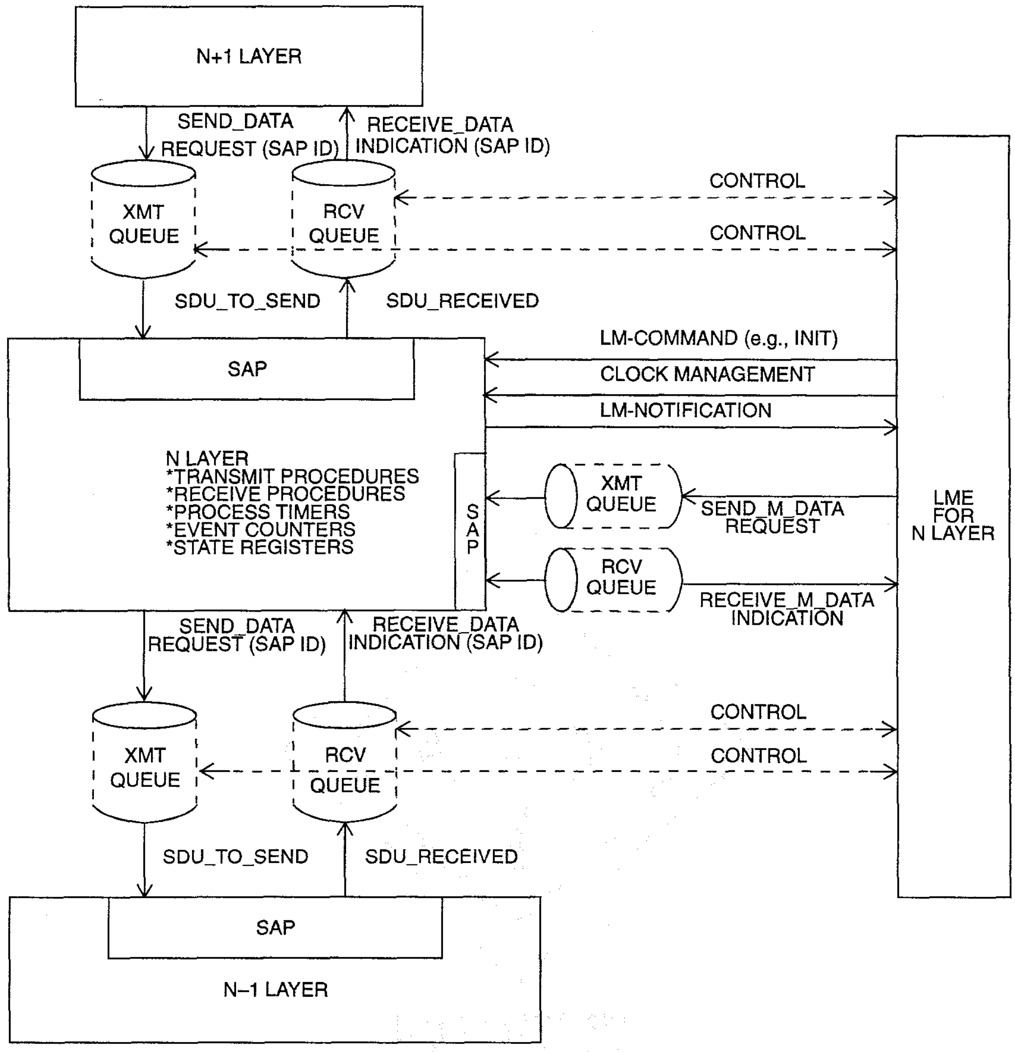


Figure 1-5—Service queues between adjacent layers

This definition of queues is independent of any particular interface implementation, and is not subject to conformance testing requirements.

1.3.3 Coordination function

The ISDN out-of-band signalling operations may be seen as being implemented by a "control" plane OSI stack that is associated with a user plane in which end-to-end user information may be conveyed. In the case of on-demand services (e.g., a dialed telephone service) a sequential process of call establishment, followed by an information interchange phase (in the user plane), and finally a call release phase, is required. To ensure that provision of the end-to-end user service is fully established and functional, a coordination function is required. In providing the dynamic bandwidth allocation functions provided in the IEEE 802.9 PHY Layer's HMUX sublayer, there is a need for additional higher (OSI) level coordination functions. The details of these functional requirements can be found in Chapter 8.

1.3.4 State machine notation

The operation of a protocol can be described by subdividing the protocol into a number of interrelated functions. The operation of a function can be described by state machines. Each machine represents the domain of a function and consists of a group of connected, mutually exclusive states. The specification of the state machine determines the order of the transition. Only one state of a function is active at any given time.

Signals and flags refer to the form by which one functional block communicates with another. Both terms arise from the use of the state machines as a descriptive method in this standard. Signals are transient communications (in theory, they are of infinitesimal width) that occur during a change of state. Flags are steady, nontransient indications that remain constant until explicitly changed.

Each state that the function can assume is represented by a box or a circle. All permissible transitions between states of a function are represented by arrows from one state to the next state. The condition that causes a change from one state to another is shown next to the transition arrow.

1.3.5 General introduction of finite state machine (FSM) diagrams and tables

The procedures employed in the receive and transmit functions that operate to convey the request/grant protocol are specified in terms of a finite state machine (FSM). The principles of FSM are as follows:

a) A process is evaluated as one finite step during an instant in time.

b) Each "state" represents an information level consisting of:

   1) *Output operand directives (commands).* These output commands specify the work output from the FSM.

   2) *Input operand directives (test events/signals).* These input events are utilized by the FSM process to determine what state transition should be performed. These input events may be subdivided into external events (signals) and internal events (signals).

c) A description of a process may be expressed via a combination of states, control flags/state variables, input events (test conditions), and output actions.

d) This FSM definition technique is very implementation styled since it defines things in such fine detail.

e) The FSM documentation consists of two forms of graphic documentation:

   1) FSM state diagrams

   2) FSM state tables

1.3.5.1 FSM diagrams

The FSM diagrams are intended to illustrate a "minimum" set of processing states that comprise the functional operation of the process. Figure 1-6 illustrates the manner in which the FSM diagrams are used in this standard.

In figure 1-6, the FSM has two states: A and B. The occurrence of External Signal 1 equals "true" will cause a transition from State A to State B. Once the FSM is operational in State B, the occurrence of Internal Signal 2 equals "false" will cause the FSM to transition back to State A. Note that for the purpose of documentation, the value "0" will be equivalent to the condition "false" and the value "1" will be equivalent to the condition "true."

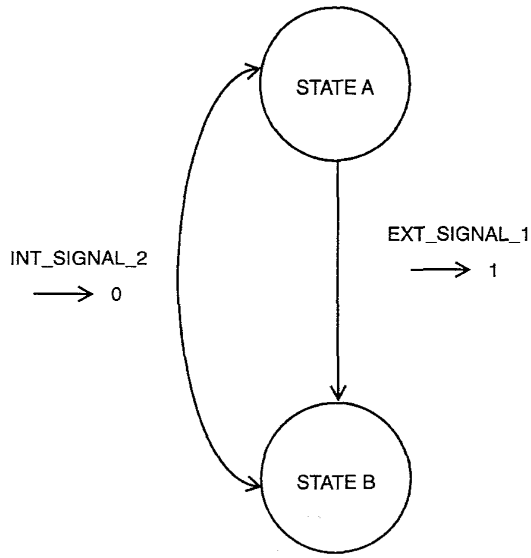


Figure 1-6—Example FSM diagram

As can be seen from this example, the FSM diagram technique overviews the essential operational states and identifies the key signal events that cause a transition from one state to another. In order to illustrate the precise details of input signal processing, output signal generation, and the treatment of error counters/timers/control variables, there is a companion description called state transition tables.

1.3.5.2 FSM state transition tables

FSM state transition tables are a companion description to the FSM state diagrams. These tables illustrate a finite set of actions that occur while the FSM is operational within a given state. Table 1-1 illustrates an example of a portion of a state transition table that would accompany the state diagram in figure 1-6.

Table 1-1—Example state transition table

Current state	Event	Action(s)	Next state
STATE_A	EXT_SIGNAL_1→1	SEND_MESSAGE_2 SET_FLAG_3:=1 START_TIMER_1	STATE_B
STATE_B	INT_SIGNAL_2→0	SEND_MESSAGE_4 SET_FLAG_3:=0 STOP_TIMER_1	STATE_A

In this example, a horizontal row represents a finite FSM processing step. While the FSM is in State A, it will regularly test or sample for the presence of external signal EXT_SIGNAL_1. When this EXT_SIGNAL_1 transitions to a "1," this will cause the FSM to vector to State B. In addition, the output actions will be to

— Send MESSAGE_2
— Set internal flag FLAG_3 to value "1"
— Start timer TIMER_1

While the FSM is in State B, it will regularly test for the presence of internal signal INT_SIGNAL_2. When this signal transitions to a "0," this will cause the FSM to vector back to State A. In addition, the output actions will be to

— Send MESSAGE_4
— Clear internal flag FLAG_3 to value "0"
— Stop timer TIMER_1

1.4 Document organization

This standard is organized as follows:

Chapters 1 through 4 contain the overview, references, definitions, and abbreviations, respectively.

Chapter 5 presents the overall architecture of the IEEE 802.9 ISLAN. The protocol sublayers and the layer service boundaries are introduced.

Chapter 6 presents the specification of the structure for the TDM and MAC frames. The individual fields are defined.

Chapter 7 presents the service specifications for the LLC/MAC and the MAC/PHY for the P channel. The isochronous service specifications from PHY to Layer 2 are presented. The management service specifications for all sublayers are presented.

Chapter 8 presents a detailed description of the procedures for the various functional modules that operate within the MAC. The descriptions contain detailed FSM models to describe the functional operations.

Chapter 9 provides a detailed description of the three sublayers of the PHY Layer: the HMUX sublayer, the PS sublayer, and the physical medium dependent (PMD) sublayer. The IEEE 802.9 interface supports two PMD sublayers: a 4.096 Mb/s line rate interface and a 20.480 Mb/s line rate interface. Complete descriptions for both are contained in this chapter.

Chapter 10 presents the ISLAN layer management. A complete description of the managed objects (MOs) contained in the management information base (MIB) for the IEEE 802.9 interface is presented.

Chapter 11 presents ISDN signalling and management, followed by a complete description of the information flow of ISDN management information. A set of enhancements to the ISDN standard repertoire is presented to complete the signalling message set.

Annex A contains the Protocol Implementation Conformance Statement (PICS) proforma. The detailed specification of the PICS is a matter for further study.

Annex B contains the Managed Object Conformance Statement (MOCS). The detailed specification of the MOCS is a matter for further study.

Annex C contains GDMO specifications of information elements required for the full definition of ISLAN systems and layer management.

Annex D describes the CCITT Q.93x message set used during local bandwidth management between the AU and the ISTE. The relationship of the HMC control field exchange between the AU and the ISTE and their respective layer management entities are defined.

Annex E provides reference models for the AU and guidelines for implementation.

Annex F contains the SDL description of the information flow across layers.

Annex G describes the optional procedures for secure control involved with the RSC procedures used to gate on/off an SCD that resides at the network side of the AU.

Annex H describes how address structures may be applied for a topology involving addressing techniques to access a wide area network (WAN) destination.

Annex I describes the optional procedures involved with the support of a cell bearer service across an IEEE 802.9 subnetwork. This service results in a restructure of the MAC frame so that the cell-based payload units may be transported.

2. References

The IEEE 802.9 ISLAN standard's architecture is required to interwork with the standard data-only and integrated services local area network (LAN) and metropolitan area network (MAN) backbone networks, as well as with ISDN networks. As such, the following specifications are important references:

ANSI/EIA/TIA-568-91, Commercial Building Telecommunications Wiring Standard.[2]

ANSI X3.139-1987 (Reaff 1992), Information Systems—Fiber Distributed Data Interface (FDDI)—Token Ring Media Access Control (MAC).[3]

ANSI X3.186-1992, Fiber Distributed Data Interface (FDDI) Hybrid Ring Control (HRC).

CCITT Recommendation E.164 (1991), Numbering plan for the ISDN era, Rev. 1 (I.331).[4]

CCITT Recommendation F.69 (1993), Plan for telex destination codes, Rev. 1.

CCITT Recommendation I.112 (1993), Vocabulary of terms for ISDNs, Rev. 1.

CCITT Recommendation I.210 (1993), Principles of telecommunication services supported by ISDN and the means to describe them, Rev. 1.

CCITT Recommendation I.320 (1988), ISDN protocol reference model. In vol. III.8 of the *CCITT Blue Books*—Integrated Services Digital Network (ISDN)—Overall network aspects and function, user-network interface.

CCITT Recommendation I.412 (1988), ISDN user-network interfaces—Interface structures and access capabilities. In vol. III.8 of the *CCITT Blue Books*—Integrated Services Digital Network (ISDN)—Overall network aspects and function, user-network interface.

CCITT Recommendation I.430 (1993), Basic user-network interface—Layer 1 specification, Rev. 1.

CCITT Recommendation I.431 (1993), Primary rate user-network interface—Layer 1 specification, Rev. 1.

CCITT Recommendation I.440 (Q.920 protocol) (1993), ISDN user-network interface data link layer—General aspects, Rev. 1.

CCITT Recommendation I.441 (Q.921 protocol) (1993), ISDN user-network interface—Data link layer specification, Rev. 1.

CCITT Recommendation I.450 (Q.930 protocol) (1993), ISDN user-network interface layer 3—General aspects, Rev. 1.

CCITT Recommendation I.451 (Q.931 protocol) (1993), ISDN user-network interface layer 3 specification for basic call control, Rev. 1.

[2]EIA publications are available from Global Engineering, 1990 M Street NW, Suite 400, Washington, DC, 20036, USA.

[3]ANSI publications are available from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

[4]All CCITT publications are now being labeled as International Telecommunications Union (ITU-T) publications. They are available from the International Telecommunications Union, Sales Section, Place des Nations, CH-1211, Genève 20, Switzerland/Suisse. They are also available in the United States from the U.S. Department of Commerce, Technology Administration, National Technical Information Service (NTIS), Springfield, VA 22161, USA.

CCITT Recommendation I.452 (Q.932 protocol) (1993), Generic procedures for the control of ISDN supplementary services, Rev. 1.

CCITT Recommendation Q.933 (1993), Layer 3 signalling specification for frame mode bearer service.

CCITT Recommendation Q.940 (1988), ISDN user-network interface protocol for management—General aspects. In vol. VI.11 of the *CCITT Blue Books*—Digital access signalling system, network layer, user—network management.

CCITT Recommendation X.3 (1993), Packet assembly disassembly facility (PAD) in a public data network, Rev. 1.

CCITT Recommendation X.25 (1993), Interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for terminals operating in the packet mode and connected to public data networks by dedicated circuit, Rev. 1.

CCITT Recommendation X.28 (1993), DTE/DCE interface for a start-stop mode data terminal equipment accessing the packet assembly/disassembly facility (PAD) in a public data network situated in the same country, Rev. 1.

CCITT Recommendation X.29 (1993), Procedures for the exchange of control information and user data between a packet assembly/disassembly (PAD) facility and a packet mode DTE or another PAD, Rev. 1.

CCITT Recommendation X.121 (1992), International numbering plan for public data networks, Rev. 1.

CCITT Recommendation X.200 (1988), Reference Model of Open Systems Interconnection for CCITT Applications. In vol. VIII.4 of the *CCITT Blue Books*—Open Systems Interconnection (OSI)—model and notation, service definition.

CCITT Recommendation X.212 (1988), Data link service definition for open systems interconnection for CCITT applications. In vol. VIII.4 of the *CCITT Blue Books*—Open Systems Interconnection (OSI)—model and notation, service definition.

CCITT Recommendation Z.100 (1993), Specification and description language (SDL), Rev. 1.

ECMA TR 44: 1989, An Architectural Framework for Private Networks.[5]

ECMA TR 51: 1990, Requirements for Access to Integrated Voice and Data Local and Metropolitan Area Networks.

EIA SP-2840A, Commercial Building Telecommunications Cabling Standard, April 19, 1994 (Draft).[6]

IEC 60-1 (1989), High-voltage test techniques—Part 1: General definitions and test requirements.[7]

IEC 60-3 (1976), High-voltage test techniques—Part 3: Measuring devices.

IEC 60-4 (1977), High-voltage test techniques—Part 4: Application guide for measuring devices.

[5]ECMA publications are available from Global Engineering, 1990 M Street NW, Suite 400, Washington, DC, 20036, USA.

[6]EIA publications are available from Global Engineering, 1990 M Street NW, Suite 400, Washington, DC, 20036, USA. This standard is expected to be published in February 1995.

[7]IEC publications are available from IEC Sales Department, Case Postale 131, 3 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse. IEC publications are also available in the United States from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

IEC 955: 1989, Process data highway, Type C (PROWAY C), for distributed process control systems.

IEEE Std 802-1990, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture (ANSI).[8]

IEEE Std 802.1B-1992, IEEE Standards for Local and Metropolitan Area Networks: LAN/MAN Management (ANSI).

IEEE Std 802.1F-1993, IEEE Standards for Local and Metropolitan Area Networks: Common Definitions and Procedures for IEEE 802 Management Information (ANSI).

ISO 3166: 1993, Codes for the representation of names of countries.[9]

ISO 6523: 1984, Data interchange—Structures for the identification of organizations.

ISO 7498-4: 1989, Information processing systems—Open Systems Interconnection—Basic Reference Model— Part 4: Management framework.

ISO 8648: 1988, Information processing systems—Open Systems Interconnection—Internal organization of the Network Layer.

ISO 8824: 1990, Information technology—Open Systems Interconnection—Specification of Abstract Syntax Notation One (ASN.1).

ISO/IEC 8208: 1990, Information technology—Data communications—X.25 Packet Layer Protocol for Data Terminal Equipment.

ISO/IEC 8348: 1993, Information technology—Open Systems Interconnection—Network Service Definition.

ISO/IEC 8802-2: 1994 [IEEE Std 802.2, 1994 Edition], Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 2: Logical link control.

ISO/IEC 8802-3: 1993 [ANSI/IEEE Std 802.3, 1993 Edition], Information technology—Local and metropolitan area networks—Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications.

ISO/IEC 8802-4: 1990 [ANSI/IEEE Std 802.4-1990], Information processing systems—Local area networks—Part 4: Token-passing bus access method and physical layer specifications.

ISO/IEC 8802-5: 1992 [ANSI/IEEE Std 802.5-1992], Information technology—Local and metropolitan area networks—Part 5: Token ring access method and physical layer specifications.

ISO/IEC 8802-6: 1994 [ANSI/IEEE Std 802.6, 1994 Edition], Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 6: Distributed Queue Dual Bus (DQDB) access method and physical layer specifications.

[8]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA.

[9]ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse. ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

ISO/IEC 8877: 1992, Information technology—Telecommunications and information exchange between systems—Interface connector and contact assignments for ISDN Basic Access Interface located at reference points S and T.

ISO/IEC 8886: 1992, Information technology—Telecommunications and information exchange between systems—Data link service definition for Open Systems Interconnection.

ISO/IEC 9506: 1991, Industrial automation systems—Manufacturing message specification.

ISO/IEC 9595: 1991, Information technology—Open Systems Interconnection—Common management information service definition.

ISO/IEC 9596-1: 1991, Information technology—Open Systems Interconnection—Common management information protocol—Part 1: Specification.

ISO/IEC 9596-2: 1993, Information technology—Open Systems Interconnection—Common management information protocol: Protocol Implementation Conformance Statement (PICS) proforma.

ISO/IEC 10022: 1990, Information technology—Open Systems Interconnection—Physical Service Definition.

ISO/IEC 10038: 1993 [ANSI/IEEE Std 802.1D, 1993 Edition], Information technology—Telecommunications and information exchange between systems—Local area networks—Media access control (MAC) bridges.

ISO/IEC 10039: 1991, Information technology—Open Systems Interconnection—Local area networks—Medium Access Control (MAC) service definition.

ISO/IEC 10040: 1992, Information technology—Open Systems Interconnection—Systems management overview.

ISO/IEC 10164-1: 1993, Information technology—Open Systems Interconnection—Systems Management: Object Management Function.

ISO/IEC 10164-2: 1993, Information technology—Open Systems Interconnection—Systems Management: State Management Function.

ISO/IEC 10164-3: 1993, Information technology—Open Systems Interconnection—Systems Management: Attributes for representing relationships.

ISO/IEC 10164-4: 1992, Information technology—Open Systems Interconnection—Systems Management: Alarm reporting function.

ISO/IEC 10164-5: 1993, Information technology—Open Systems Interconnection—Systems Management: Event Report Management Function.

ISO/IEC 10164-6: 1993, Information technology—Open Systems Interconnection—Systems Management: Log control function.

ISO/IEC 10164-7: 1992, Information technology—Open Systems Interconnection—Systems Management: Security alarm reporting function.

ISO/IEC 10164-8: 1993, Information technology—Open Systems Interconnection—Systems Management: Security audit trail function.

ISO/IEC 10165-1: 1993, Information technology—Open Systems Interconnection—Management Information Services—Structure of management information: Management Information Model.

ISO/IEC 10165-2: 1992, Information technology—Open Systems Interconnection—Structure of management information: Definition of management information.

ISO/IEC 10165-4: 1992, Information technology—Open Systems Interconnection—Structure of management information—Part 4: Guidelines for the definition of managed objects.

ISO/IEC 15802-4: 1994 [ANSI/IEEE Std 802.1E, 1994 Edition], Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Common specifications—Part 4: System load protocol.

ISO/IEC DIS 10165-5...[10] Information technology—Open Systems Interconnection—Structure of management information: Generic management information.

ISO/IEC DIS 15802-2...[11] [IEEE Std 802.1B and 802.1k], Information technology—Telecommunications and information exchange between systems—IEEE Standards for Local and Metropolitan Area Networks—Common specifications—Part 2: LAN/MAN management, service and protocol.

ISO/IEC TR 9577: 1993, Information technology—Telecommunications and information exchange between systems—Protocol identification in the network layer.

ISO/IEC TR 10178: 1992, Information technology—Telecommunications and information exchange between systems—The structure and coding of Logical Link Control addresses in Local Area Networks.

[10]Presently at state of Draft International Standard.

[11]See footnote 10.

3. Definitions

The following are definitions of terms used in this standard that are widely used in IEEE 802 as well as CCITT I series and other standards documents. For the CCITT Recommendation I series, the basic vocabulary and terms can be found in CCITT Recommendation I.112 (1993).[12]

3.1 access unit (AU): The abstraction of the device that provides the IEEE 802.9 functionality, i.e., the integrated set of services, to stations connected across the IEEE 802.9 interface.

3.2 bearer service: A telecommunication service that provides the capability for the transmission of signals between user-network interfaces [see CCITT Recommendation I.210 (1993)].

3.3 bearer channel protocol intervention level: The highest protocol level at which a private switching network (PSN) provides protocol termination on a given bearer channel.

3.4 B channel: A channel that provides 64 kb/s, full duplex, isochronous access. B channels support all ISDN bearer services. The information on a B channel may be non-switched or either circuit or packet switched depending on user request and network capabilities. In the IEEE 802.9 TDM frame, two B channel slots have been defined. There may be more than two B channels configured across a given IEEE 802.9 interface (in this event the B channels would be configured as $C sub 1$ channels).

3.5 C channel: A channel that provides an integer multiple of 64 kb/s, full duplex, isochronous clear channels. The "C" is used to indicate that this is a circuit-switched channel. $C sub m$ stands for a channel (m) times 64 kb/s in size. In general, a C channel has the same characteristics as an ISDN B or H channel, except that it can be any multiple of 64 kb/s in size, rather than only the CCITT approved rates. There may be multiple C channels of varying sizes (n × C).

Note the following equivalencies:

C_1	$= B$	= 64 kb/s
C_6	$= H_0$	= 384 kb/s
C_{24}	$= H_{11}$	= 1.536 Mb/s
C_{30}	$= H_{12}$	= 1.920 Mb/s

The maximum composite bandwidth allocated for the C channel(s) shall be limited by the application requirements for the P channel. That is, the available bandwidth that may be assigned for C channels shall be limited to the difference between the TDM payload field minus the bandwidth required for the P channel. In practice, the C channel size(s) will be effected by the interconnection facility(ies) between the AU and the ISDN wide area network (WAN).

3.6 convergence protocol: A protocol that provides the convergence service for the provision of enhancements to an underlying service in order to provide for the specific requirements of the convergence service user.

3.7 coordinating entity: That part of the Network Layer within an end system or interworking unit responsible for the coordination and synchronization of functions belonging to the Data and Signalling subentities of the layer entity implementing a PSN access protocol.

3.8 D channel: A channel that provides 16 kb/s or 64 kb/s, full duplex packet access. The D channel provides ISDN call control services via the CCITT Q.93x family of protocols. The use of the D channel may be restricted to conveying signalling information in some applications. The D channel is capable of supporting packet mode bearer information. All information on the D channel is packetized.

[12]Information on references can be found in Chapter 2.

3.9 data circuit terminating equipment (DCE): A device that provides the signal conversion and coding between the data terminal equipment and the network carrier facility. Note that in the context of a CCITT X.25 network, for example, the DCE performs functions at the network end of an access line to the network.

3.10 data terminal equipment (DTE): A device that serves as a data source and/or a data sink.

3.11 H channel: A wideband channel (i.e., a channel that contains multiples of 64 kb/s). See CCITT Recommendation I.412 (1988).

3.12 in-band signalling: Signalling applications in which the signalling information is transmitted in the same information flow as the data. *See also:* **out-of-band signalling**.

3.13 integrated services terminal equipment (ISTE): A device that serves as an information source and/or an information sink for the provision of voice, facsimile, video, data, and other information.

3.14 interworking unit (IWU): A unit that provides the functions needed to allow interworking between a PSN and another network, e.g., interworking between a PSN and a CCITT X.25 packet-switched public data network (PSPDN).

3.15 isochronous: A communication stream transport that is uniform in time. The delivery of the physical stream of information is recurring at regular intervals.

3.16 layer service: The service that a layer provides to the next higher layer. See ISO 7498-4: 1989 and CCITT Recommendation X.200 (1988).

3.17 out-of-band signalling: Signalling applications in which the signalling information is outside of the user information channel, whether or not transmitted in a different physical or logical channel from the associated user data, e.g., over different physical paths, in different time-slots in a time division multiplex (TDM) stream.

3.18 P channel: A full duplex packet channel that provides IEEE 802 MAC Layer services. The P channel may optionally carry CCITT Q.93x (where x refers to the family of CCITT Q.930 protocols) call control in a logically out-of-band fashion, which may be highlighted by referring to it as P_D. Presently there is only one P channel defined per ISLAN interface. The minimum size of the P channel shall be limited by the application requirements of the topology that represents the connection of the AU to the backbone LAN application. The support of station management communication will be an important factor in the characterization of P channel bandwidth size.

3.19 packet assembler/disassembler (PAD): A device that assembles and disassembles packets. Examples are found in CCITT Recommendations X.3 (1993), X.28 (1993), and X.29 (1993).

3.20 packet handler (PH): A device for processing packets or frames in a manner so as to be able to route individual frames or packets out of one data stream into multiple different data streams.

3.21 packet mode protocol: The access protocol based on the use of out-of-band control and Link Layer multiplexing. *Syn:* PSN packet mode protocol.

3.22 packet switched public data network (PSPDN): A public subnetwork that is accessed via the CCITT X.25 protocol and that provides virtual circuit service.

3.23 packetized data transfer: Transfer of data through a network where data is conveyed in packets and/or frames in a statistical manner. Packets or frames are propagated through the network and delivered to destinations based on addressing information contained therein.

3.24 private packet/frame and circuit switching network (PPCSN): A private switching network that provides both circuit and packet/frame switching functions (i.e., all the functions of both PCSNs and PPSNs).

3.25 private circuit switching network (PCSN): A private network that only provides circuit switching functions, except that it may be able to transport packetized "user-to-user" information passed over the signalling channel.

3.26 private packet/frame switching network (PPSN): A private network that only provides packet/frame switching functions.

3.27 private switching network (PSN): A private network that provides switching functions (circuit and/or packet/frame switching). It is operated by the user and located on user premises to cover the communications needs in the user's domain. The term private-switching network includes both the private circuit-switching network and the private packet-switching network.

3.28 packet-switched data network (PSDN): A packet-switched subnetwork that can be a PPSN (private) or a PSPDN (public) packet-switched network.

3.29 R interface: The interface provided at the R reference point to allow the connection of non-ISDN terminals using, for example, CCITT V- or X-series interfaces.

3.30 relaying: A function performed at intermediate nodes on an interconnection between communicating end-systems. The relaying function is performed by connecting two independent layer entities. For example, a relaying function at the Data Link Layer connects two Data Link Layer entities to make an interconnection.

3.31 S interface: The interface provided at the S reference point for ISDN user-to-network interface.

3.32 signalling: The exchange of information specifically concerned with the establishment and control of connections, and the transfer of user-to-user and management information in a telecommunication network, e.g., in a PPSN.

3.33 subnetwork: A set of one or more intermediate open systems that provide relaying and through which end open systems may communicate. It is a representation within the OSI model of a real network such as a carrier network, a provider network, or a LAN. It can also be defined as a collection of equipment and physical media that forms an autonomous whole and that can be used to interconnect real systems for the purpose of communication (see ISO 8648: 1988).

3.34 subnetwork service: The service supported by the subnetwork access protocol.

3.35 terminal adapter (TA): An adapter required to map one specified interface to another. An example is the adapting of the R interface in the CCITT ISDN interface (e.g., according to the CCITT V- or X-series) to the S interface. The TA may be an integral functional entity as part of the terminal or may be a separate physical unit connected between an R interface and the S interface.

3.36 virtual circuit: The generic concept of a logical connection. A virtual circuit may be implemented by means of a frame-switched service or a packet-switched service.

4. Abbreviations

The following abbreviations are used in this standard as well as in IEEE 802 CCITT I series and other standards documents. For the CCITT Recommendation I series, the basic vocabulary and terms can be found in CCITT Recommendation I.112 (1993).

AC	access control field
ACSE	association control service element
ADPCM	adaptive differential pulse code modulation
AFI	authority and format identifier
AM/PM	amplitude modulation/phase modulation
APDU	application protocol data unit
APM	additional packet mode services
ANSI	American National Standards Institute
ASN.1	Abstract Syntax Notation One
ATM	asynchronous transfer mode
AU	access unit
B	B channel
BB	broadband
BCD	binary coded decimal
BER	bit error rate
BISDN	broadband integrated services digital network
BRI	basic rate interface
C	circuit-switched channel
CAP	combined amplitude and phase
CC	country code
CCITT	Consultative Committee of International Telephone and Telegraph
CEI	connection endpoint identifier
CLNP	connectionless network protocol
CMIP	common management information protocol
CMIPDU	common management information protocol data unit
CMIS	common management information service
CMISE	common management information service element
CO	central office
CRC	cyclic redundancy check
D	D channel
DA	destination address
DCC	data country code

DCE	data communication equipment
DCR	D channel rate select bit
DIW	D-type inside wire
DL	data link layer
DLC	data link control
DSAP	destination service access point
DSP	domain-specific part
DSS	digital signalling system 1
DTE	data terminal equipment
ECMA	European Computer Manufacturers Association
EMI	electromagnetic interference
FAW	frame alignment word
FC	frame control
FCC	Federal Communications Commission
FCS	frame check sequence
FDDI	fiber distributed data interface
FIFO	first-in first-out
FSM	finite state machine
G	grant
GDMO	guidelines for the definition of managed objects
GOSIP	Government OSI profile
H	H channel
HDLC	high-level data link control
HMC	hybrid multiplexer control field
HMF	hybrid multiplexer multiframing bit
HMUX	hybrid multiplexer
ICD	international code designator
IDI	initial domain identifier
IDP	initial domain part
IE	information element
IEC	International Electronic Commission
IP	internet protocol
IPX	internet protocol X
IS	integrated services
ISDN	integrated services digital network
ISLAN	integrated services LAN

ISO	International Standards Organization
ISPBX	integrated services private branch exchange
ISTA	integrated services terminal adaptor
ISTE	integrated services terminal equipment
ITU-T	International Telecommunication Union–Telecommunication Standardization Sector
IWU	interworking unit
LAN	local area network
LAPB	link access procedure, balanced
LAPD	link access procedures for the D channel
LAPD+	extended procedures for LAPD
LEN	length field
LLA	logical link address
LLB	local loopback status bit
LLC	logical link control
LLS	logical link services
LM	layer management
LME	layer management entity
LMI	layer management interface
LMMS	layer management MAC services
LMMP	layer management MAC protocol
LPDU	logical link protocol data unit
LSAP	link layer service access point
LSb	least significant bit
LSB	least significant byte
LSDU	link layer service data unit
LTB	local template buffer
MA	management
MAC	medium access control
MAN	metropolitan area network
MAPDU	management application protocol data unit
MDI	medium dependent interface
MIB	management information base
MIS	management information services
MLME	managed layer management entity
MO	managed object
MOCS	managed object conformance statement

MPDU	medium access control protocol data unit
MSAP	medium access control service access point
MSb	most significant bit
MSB	most significant byte
MSDU	MAC service data unit
MT	management
MUX	multiplexing
NDC	national destination code
NOT	notification
NSAP	network service access point
NSN	national significant number
NTW	network
OP	operation
OSI	Open Systems Interconnection
P	packet channel
PABX	private automated branch exchange
PAD	packet assembler disassembler
PAR	project authorization request
PBX	private branch exchange
PCM	pulse code modulation
PDN	packet data network
PDU	protocol data unit
PERR	parity error status bit
PEVN	even parity cumulative bit
PhSAP	Physical Layer service access point
PH	packet handler
PHY	Physical Layer
PLME	Physical Layer management entity
PLS	Physical Layer signalling
PMA	physical medium attachment
PMD	physical medium dependent
PODD	odd parity cumulative bit
PCSN	private circuit switching network
PPCSN	private packet/frame and circuit switching network
PPE	provider protocol entity
PRI	primary rate interface

PS	physical signalling
PSPDN	packet-switched public data network
PSN	private switching network
PTT	Post, Telegraph, and Telecommunications (administration)
Q.93x	CCITT Q series protocol ("x" refers to the members of the CCITT Q.930 family of protocols)
QOS	quality of service
R	request
RLBREQ	remote loopback request bit
RLBRESP	remote loopback response bit
RSC	remote secure control
ROSE	remote operation service element
SA	source address
SAP	service access point
SAPI	service access point identifier
SB	service boundary
SCD	secure communications device
SDL	sequence and description language
SDLC	synchronous data link control
SDU	service data unit
SID	service identifier field
SMAE	system management application entity
SMAP	system management application process
SMASE	system management application service element
SMI	structure of management information
SN	subscriber number
SNMP	simple network management protocol
SNPA	subnetwork point of attachment
SOF	start-of-frame bit
SSAP	source service access point
SYN	synchronization field
TA	terminal adaptor
TCP/IP	transport control protocol/Internet protocol
TDM	time division multiplexer
TDMMTN	time division multiplexer maintenance field
TE	terminal equipment

TEI	terminal endpoint identifier
TTP	telephone twisted pair
TXC	template exchange notification bit
UTTP	unshielded telephone twisted pair
UPE	user provider entity
VCCI	virtual cell circuit identifier
WAN	wide area network
WG	working group

5. Architecture

5.1 Overview and basic concepts

The ISLAN interface is an integrated voice, data, and video interface that provides packet service and isochronous digital channels on a full duplex interface to the desktop over unshielded telephone twisted-pair (UTTP) wiring. The principal target of the initial Physical (PHY) Layer specification is the office environment on premises-based networks. It is intended to be used on UTTP wiring in a physical point-to-point configuration.

The integrated services are provided to the IS terminal equipment (ISTE) across an interface serviced by a functional entity called the access unit (AU). The ISTEs are connected by UTTP wiring to the AU. Thus, from the perspective of the ISTEs, their PHY interface is to the AU and the services are functionally provided via the AU. From the perspective of the ISTE, there is one interface into which it sends/receives a multiplexed communication stream, independent of the network topology. The AU accommodates two fundamentally different application topologies. First, the interface may be to a stand-alone LAN in which the AU actually provides the complete pathway for the required services of the attached ISTEs. The second application is the environment wherein the IEEE 802.9 ISLAN serves an access interface that feeds into a backbone consisting of an IEEE 802.x LAN, a fiber distributed data interface (FDDI), or an integrated services digital network (ISDN) provided via a private network using a PABX or via the public network. In this case, the AU becomes an interworking unit to the system providing the services. Combinations of the two scenarios are also possible. Unlike the IEEE 802.3/4/5/6 LANs, the medium between the ISTE and the AU is not shared; rather, it is a point-to-point configuration.

The scope of this standard includes the access link between the IEEE 802.9 ISTE and the AU. Figure 5-1 provides an architectural overview of the IEEE 802.9 ISLAN interface to better explain how the MAC arises, and what its point of departure is from the other IEEE 802 MACs. Figure 5-1 is a logical view only and does not imply any particular implementation. The connection between the AU and the ISTEs is a star arrangement. Carried over these point-to-point links between the AU and the ISTEs is a multiplexed bit stream of packet data, voice, wideband data, and other services such as facsimile, image, or video delivered over isochronous channels. The ISTEs may contend for access to integrated services from the AU. These services may be provided directly from the AU or from beyond the AU by networks which provide IEEE 802 LAN service, integrated services, or ISDN bearer services. Thus, beyond the AU there may exist a connection to another IEEE 802 LAN or FDDI network, an ISDN switch, or an integrated services network such as IEEE 802.6. This is illustrated in figure 5-2.

In the IEEE 802.9 architecture, the medium is not a shared resource. However, the AU is a shared resource which manages and supports multiple ISTEs, each connected in a point-to-point arrangement with the AU. In fact, the AU architecture participates as the integral manager of the shared service and directly influences how the shared resources become available. The particular AU architecture will provide the arbitration scheme for access to the medium. Note that the definition of the AU architecture is outside of the scope of this standard.

The IEEE 802.9 ISLAN architecture employs a more sophisticated PHY Layer than other LANs, and is able to support the multiplexing of several channels of information by means of a time division multiplexed (TDM) bearer. Architecturally this bearer is specified to be capable of supporting information transfer rates that are multiples of 4.096 Mb/s in isochronous frames delivered at 8 kHz. These frame are synchronous with any "host" ISDN to which the backbone side of the IS-AU may be attached. In this standard, specifications are provided for the use of 4.096 Mb/s and 20.48 Mb/s TDM bearers, having 64 and 320 slots, each comprising one octet.

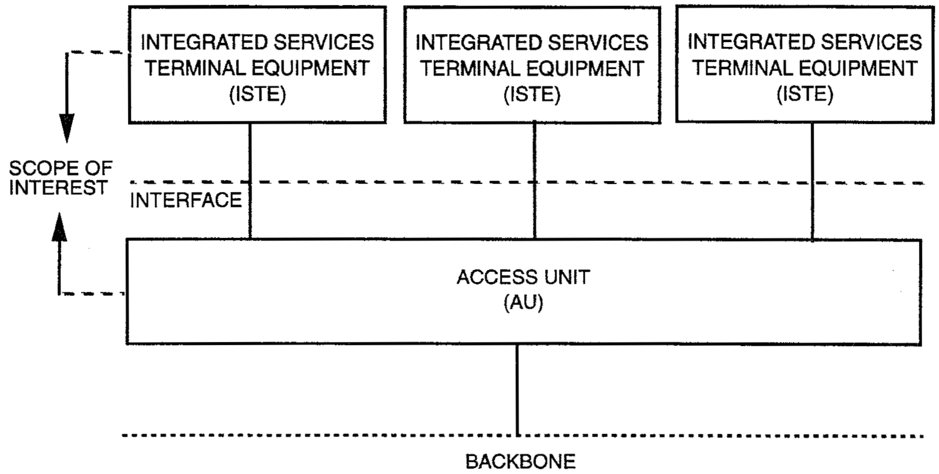


Figure 5-1—Overview of IEEE 802.9 ISLAN interface

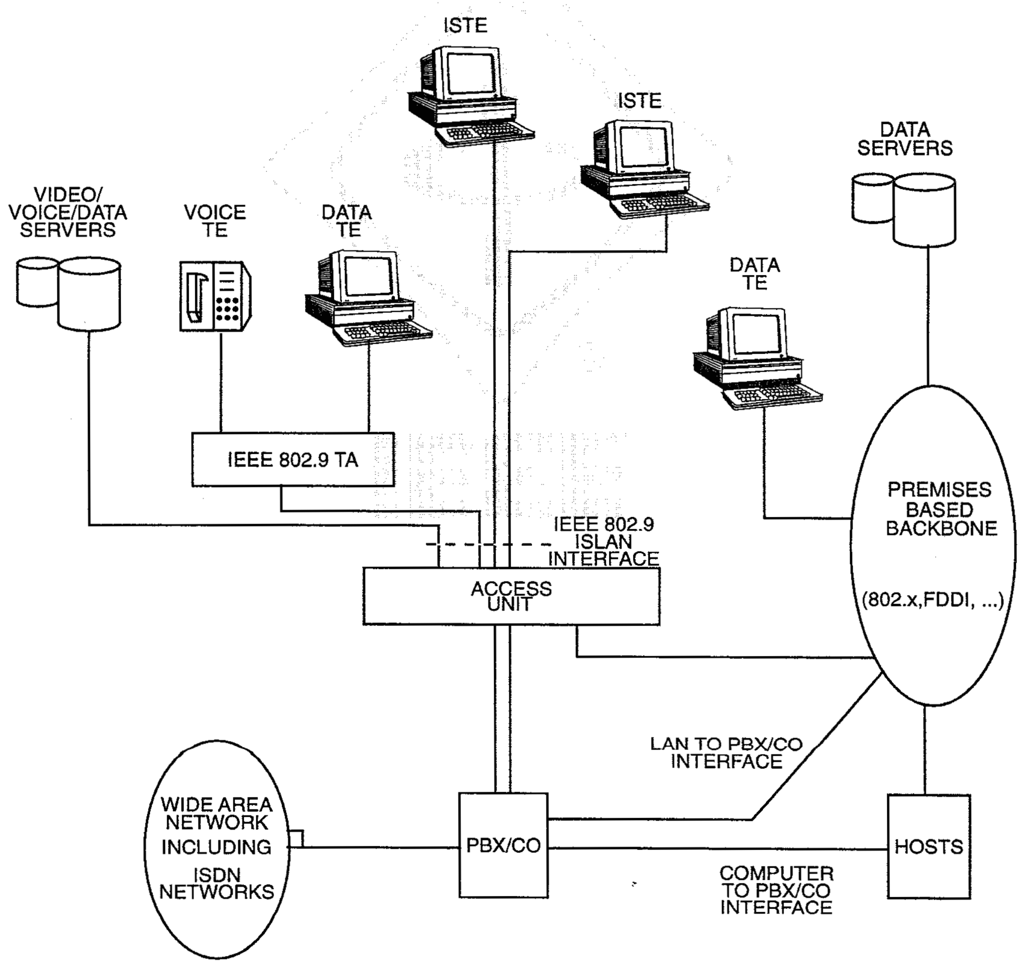


Figure 5-2—IEEE 802.9 ISLAN interface configuration

The multiplexing of the several services carried by octets or multiples thereof within the common TDM bearer is a function of the Hybrid Multiplexing (HMUX) sublayer of the IEEE 802.9 PHY Layer, but it does effect the complete management and control of the channels. HMUX sublayer management procedures combined with extension to ISDN signalling (recommended in annex D) complete the provision of a mechanism that can transparently and dynamically establish and release multiple independent user service channels. The IEEE 802.9 PHY Layer is able to offer each of these to higher level Layer 2 processes as variable bandwidth isochronous bearers.

These channel types are:

— *D channel:* A 16 kb/s or 64 kb/s, full duplex packet channel for the provision of call control service via the CCITT Q.93x family of protocols. The use of the D channel may be restricted to conveying signalling information in some applications. The D channel is capable of supporting packet mode bearer information. All information on the D channel is packetized.

— *B channel:* A full duplex, 64 kb/s isochronous channel. From the perspective of this standard, the B channel requires no specific upper layer protocol suite above the PHY's HMUX sublayer. The information on a B channel may be non-switched or either circuit or packet switched depending on user request and network capabilities. In the IEEE 802.9 TDM frame, two B channel slots have been defined. There may be more than two B channels configured across a given IEEE 802.9 interface (in this event, the B channels would be configured as C_1 channels).

— *C channel:* A full duplex isochronous channel whose bandwidth is a multiple of 64 kb/s. The expression "C" is used to indicate that this is a circuit-switched channel. C_m stands for a channel (m) times 64 kb/s in size. In general, a C channel has the same characteristics as an ISDN B or H channel, except that it can be any multiple of 64 kb/s in size, rather than only the CCITT approved rates. There may be multiple C channels of varying sizes (n × C).

  Note the following equivalencies:

  $C_1 = B = 64$ kb/s
  $C_6 = H_0 = 384$ kb/s
  $C_{24} = H_{11} = 1.536$ Mb/s
  $C_{30} = H_{12} = 1.920$ Mb/s

  The maximum composite bandwidth allocated for the C channel(s) shall be limited by the application requirements for the P channel. That is, the available bandwidth that may be assigned for C channels shall be bounded by the difference between the TDM payload field minus the bandwidth required for the P channel. In practice, the C channel size will be affected by the interconnection facility(ies) between the AU and the ISDN wide area network (WAN). As for the B channel, from the perspective of this standard, no MAC sublayer protocol is defined.

— *AC channel:* A full duplex, isochronous 64 kb/s channel for transporting information relating to the MAC for the request/grant arbitration procedure used in governing access to the P channel for MPDU transport.

— *P channel:* A full duplex packet channel that provides IEEE 802 MAC services for data. The IEEE 802.9 interface requires that CCITT Q.93x call control be used between the AU and the ISTE for the purpose of local bandwidth management. Conventionally, there is a D channel dedicated to this transport. In some network topologies, it may be possible to configure this as a "logically" separate D channel, which is referred to as P_D. The use of the P channel to carry CCITT Q.93x call control in a logically out-of-band fashion, which may be highlighted by referring to it as P_D, is a matter for further study.

Presently there is only one P channel defined per ISLAN interface. The minimum size of the P channel shall be limited by the application requirements of the topology that represents the connection of the AU to the backbone LAN application. The support of station management communication will be an important factor in the characterization of the P channel bandwidth size.

The IEEE 802.9 MAC is defined over the P channel. The services supported on the P channel may include both the IEEE 802 MAC data service and ISDN packet mode bearer services. The issue of supporting signalling based on the CCITT Q.93x family of standards on the P channel (P_D) is a matter for further study.

Since the function of the MAC is to provide underlying service for Logical Link PDUs (LPDUs), the characteristics of the IEEE 802.9 MAC will depend on the services required by the LPDUs. The ISDN bearer services for packet data utilize LAPB for CCITT X.25 service and LAPD for the additional packet mode service (APM), in addition to call control. Thus, in its full generality, the IEEE 802.9 MAC could support the IEEE 802 LLC, the LAPD, and the LAPB.

The LAPD service is uniquely identified by the service identification (SID) field of the MAC frame. The convergence details of the MAC frame support are a matter for further study.

The AU is provided as an architectural model to represent both the interworking interfaces to allow the extension of the IEEE 802.9 ISLAN interface service into the far side IS backbone such as LAN and ISDN networks, as well as the necessary support functions to allow concentration of multiple ISTE interfaces. The ISLAN interface part of the architecture of the AU is implementation independent while the backbone part is implementation dependent. This is discussed further in annex E.

To permit a wide range of capabilities of vendor implementation of IEEE 802.9-conformant products, the IEEE 802.9 ISLAN provides in the PHY Layer signalling the ability to support several modes of ISTE and AU interworking. Rudimentary IEEE 802.9 packet service only or ISDN basic rate interface (BRI) only devices can be supported. In addition, equipment that is capable of supporting multiple dynamic wideband channels is designated a specific mode of operation. Further details on this can be found in 9.2.3.

The IEEE 802.9 architecture also provides for the conveyance of broadband ISDN-conformant asynchronous transfer mode (ATM) cells. There are no normative specifications for such a service within this standard; however, there is a description in annex I of a set of methods by which such a service could be realized.

5.2 Application areas

A variety of applications have been identified for the various channels. The P channel may be used for the provision of LAN service to ISTEs using transport service which may traverse an ISDN or IS network to get to the LAN service. It may provide the IEEE 802 LAN service or the ISDN bearer service and also support call control using the CCITT Q.93x family of protocols. Among the end-user data applications are: file transfer between ISTE, remote terminal interactive sessions, remote database access by transaction-oriented programs, computer-aided design, and object-oriented and pixel-oriented graphic terminal interactive sessions.

The P channel is primarily intended for the provision of IEEE 802 LAN data. The potential for the channel to provide call control service via the CCITT Q.93x family of protocols and other ISDN services is a matter for further study.

The D channel provides for a 16 kb/s or 64 kb/s channel for call connection and control using the CCITT Q.93x family of protocols. This implies the transport of signalling information for the call connection/control of the B and C channels and also the transport of ISDN management information. The D channel could also be used to support other ISDN bearer services such as frame relay, frame switching, CCITT X.31, etc.

The B channel's bearer characteristics in the IEEE 802.9 environment are identical to that of the ISDN B channel. The B channels are primarily intended for the provision of PCM voice but could be used to provide any 64 kb/s (or rate-adapted to 64 kb/s) isochronous voice or data services. Among such services would be circuit-mode data supporting: unrestricted 64 kb/s information transfer, restricted 56 kb/s information transfer capability, asynchronous data using rate adaption, asynchronous data via CCITT X.3 packet assembler disassembler (PAD) and CCITT X.25, synchronous (SDLC, HDLC, etc.) via rate adaption, and facsimile support; and packet mode data intended for the ISDN network.

A C channel is a wideband isochronous channel that can support both circuit-mode and packet-mode services to which the interface provides access. A C channel whose bandwidth is configured to be 64 kb/s in size is identical to a B channel. Typical examples of C channel applications would include access to image and video services as well as access to ISDN wideband services.

To support existing terminals and stations on the ISLAN interface, use is made of terminal adaptors (TAs). Since the TA will depend on the native mode subnetwork being adapted to the IEEE 802.9 interface, a family of TAs is expected to result. This will facilitate interworking with the installed base while making it possible for such stations to make use of new IS applications.

5.3 Layers and layer service boundaries

The IEEE 802.9 ISLAN model and its relationship to the open system interconnection (OSI) reference model of the International Organization for Standardization (ISO) is illustrated in figure 5-3.

The scope of this standard covers the Data Link and PHY Layers for information transfer. It should be noted that CCITT Q.931, which is a Layer 3 protocol, is used for call control. The Data Link Layer protocols are defined over the packet channels, i.e., the D and P channels. The PHY Layer considerations apply to all of the channels—both packet and isochronous.

With respect to the Data Link Layer protocols, there are three types of information channels: the IEEE 802.9 P channel, isochronous channels, and the ISDN D channel. The Link Layer for the P channel consists of the IEEE 802.9 MAC sublayer together with the IEEE 802.2 LLC sublayer. The ISDN D channel services require use of the LAPD protocol at the Link Layer. The isochronous channels, the B and C channels, may use a Layer 2 and higher layer protocol, as appropriate, for the type of service that is being provided over the channel for the given data, voice, or video communication application. Note that isochronous service may not conform to the OSI stack of protocols (e.g., voice encoding streams).

The packet and isochronous channels are multiplexed at the HMUX sublayer. Thus, the physical frame contains the P, C, B, D, and access control (AC) channels, but the MAC described in this standard is the protocol defined for the P channel.

The Data Link and PHY Layers have an interface to management (MT). From a modeling perspective, each layer (or sublayer) has its individual layer management entity (LME). The combination of the individual LMEs interacting with the system management application process is described as the management entity.

The layers interact by way of well-defined interfaces to provide specific services.

5.3.1 Data Link Layer

The role of the Data Link Layer (DL) is to provide for the transparent and reliable transfer of data link protocol data units (PDUs) between peer data link entities. The functions provided are as follows.

a) *Frame delimiting.* The capability to indicate where the data link PDU begins and ends. This involves the recognition of a sequence of octets transmitted/received in the P channel of the IEEE 802.9 PHY Layer as a MAC frame.

b) *Transparency.* The capability to transfer a PDU regardless of content, format, or coding.

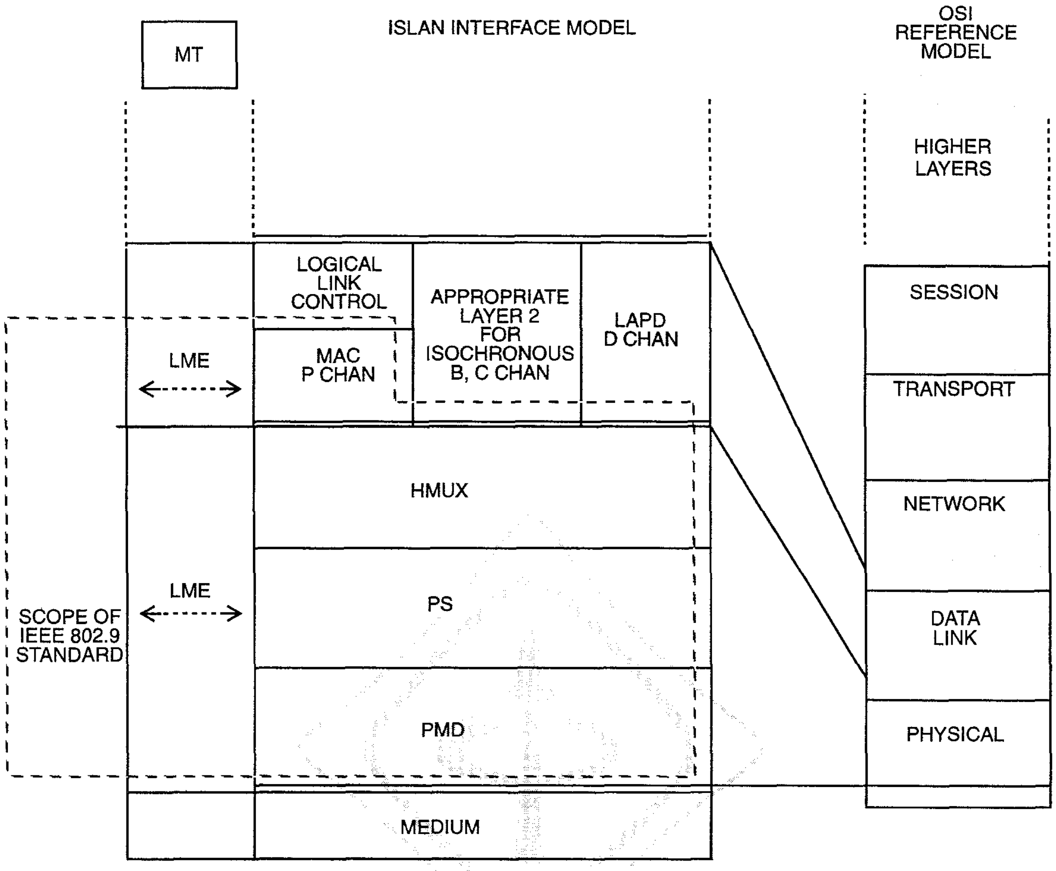


Figure 5-3—Relationship of ISLAN interface model to OSI reference model

c) *Addressing.* The capability to identify the data link service access point (SAP) to which a data link connection is established and, where multipoint links are involved, the capability to differentiate among the various connections.

d) *Error control.* The capability to detect errors and, possibly, recover from errors resulting from the physical medium. These errors include loss, corruption, insertion, and misordering.

e) *Flow control.* The capability of the receiving DL entity to regulate the rate at which the sending DL entity may send data link PDUs.

f) *Medium management.* The capability to use the physical medium for the exchange of PDUs among multiple stations. This consists of medium allocation, adaptation, and medium recovery. Medium allocation provides access to the medium for multiple stations, provides for the efficient utilization of the medium, provides for the release of the medium, and takes care of logical additions and removals of stations. Adaptation involves any scheme, such as the use of a PAD to ensure minimum frame length, to enable the proper functioning within the medium. Medium recovery restores the use of the medium following malfunction of the allocation scheme. With the star-wired topology of connecting a workstation to the AU, this recovery function becomes very simplified.

g) Procedures for the exchange of data link PDUs.

In the IEEE 802.9 model, three different types of DL services are supported:

- For the isochronous channels (B and C), the DL services are dependent on the application, and therefore this standard does not address the DL.
- For the D channel, the DL services comprise the services provided by LAPD.
- For the P channel, the DL services consist of the services defined by IEEE 802 for packet communication. The service identification (SID) field carries information that serves as a selector function to activate the appropriate MAC convergence functionality at the receiver (see 6.6.3).

In the IEEE 802 model, the DL services are provided by the MAC and the Logical Link Control (LLC) sublayers.

The relationship of the DL to the other layers and the variety of services provided are shown in figure 5-3.

5.3.1.1 LLC sublayer

The LLC sublayer constitutes the top sublayer in the DL and is common to the various medium access methods that are defined and supported by IEEE 802. It is responsible for the initiation of data link control interchange, organization of data flow, the interpretation of received command PDUs and generation of appropriate response PDUs, and actions regarding error control and error recovery functions, as needed in that sublayer. The LLC sublayer is responsible for the management of LSAP addressing. Figure 5-4 depicts the relationship between the LLC sublayer and the other layers.

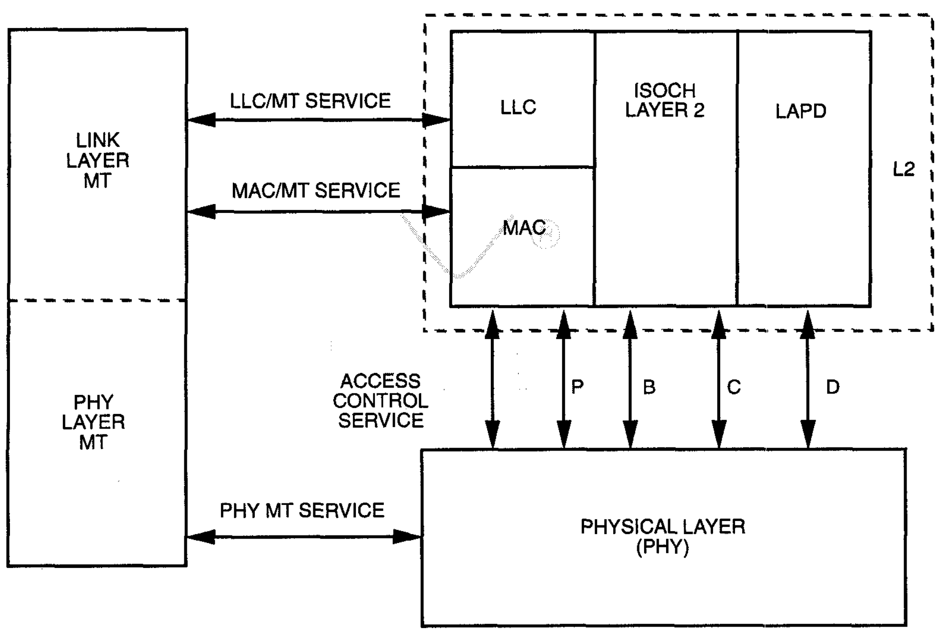


Figure 5-4—General overview of the Data Link Layer model for the IEEE 802.9 interface

5.3.1.1.1 LLC/MAC service boundary

The interface between the MAC sublayer and the LLC sublayer includes facilities for transmitting and receiving frames, and provides peer-operation status information for use by higher-level error recovery procedures.

Juniper Ex 1032 - 45
Juniper v Chrimar
IPR2016-01399

5.3.1.2 MAC sublayer

The MAC sublayer provides for the transparent transfer of LPDUs of a specified maximum size to and from the LLC sublayer. It does not restrict the content, format, or coding of the LPDU. It has interfaces to the LLC and the HMUX sublayers, as well as to management. Figure 5-4 describes the relationship between the MAC sublayer and the other layers.

The IEEE 802.9 MAC sublayer provides MAC frame delimiting, transparency, MAC addressing, error detection, and access control between the ISTEs and the AU.

5.3.1.2.1 MAC/MT service boundary

The interface between the MAC sublayer and layer management includes facilities for MT to monitor and control operations of the MAC sublayer, in the service of a system management application process (SMAP).

5.3.1.2.2 MAC/physical service boundary

The interface between the MAC sublayer and the PHY Layer includes MAC framing signals and access control information for exchanging a pair of serial bit (symbol) streams between the two layers.

5.3.1.2.3 MAC/HMUX service boundary

As shown in figure 5-5, the HMUX sublayer, in the transmit direction, receives an IEEE 802.9 MAC frame from the MAC sublayer to be carried in the P channel payload. The access is governed by the P channel access control method described in Chapter 8. The P channel access control information also passes across this boundary. In the receive direction, the P channel payload as well as the P channel access control information is passed from the HMUX sublayer to the MAC sublayer.

Both the P channel itself and the access control channel at the MAC/HMUX service boundary are isochronous bearer channels. In this respect, they are distinct from the isochronous services provided by the IEEE 802.9 PHY only in so far as they are separate channels with distinct SAPs.

5.3.1.2.4 Isochronous Layer 2/HMUX service boundary

As shown in figure 5-5, the HMUX sublayer transmits and receives C channels and B channels to and from the isochronous Layer 2.

An interface specification is provided in 9.2.4 that describes the logical signals that provide the isochronous service. These signals include information for distinguishing the transmit and receive information into separate channels, information to assist in framing (octet alignment), and timing information. To ensure a high degree of product interworking and vendor independence at the IEEE 802.9 isochronous service interface to the PHY layer, a description of the signals that implement this service is provided.

5.3.1.2.5 LAPD/HMUX service boundary

As shown in figure 5-5, the HMUX sublayer transmits and receives D channel information to and from the LAPD Layer. The D channel supports the signalling required to establish, maintain, and release CCITT ISDN call connections for bearer service.

The architectural support in IEEE 802.9 for scalable multiple independent services calls for flexible nondisruptive channel and bandwidth changes. In addition, the D channel supports extensions required to the CCITT Q.93x signalling protocols for the support of dynamic bandwidth allocation.

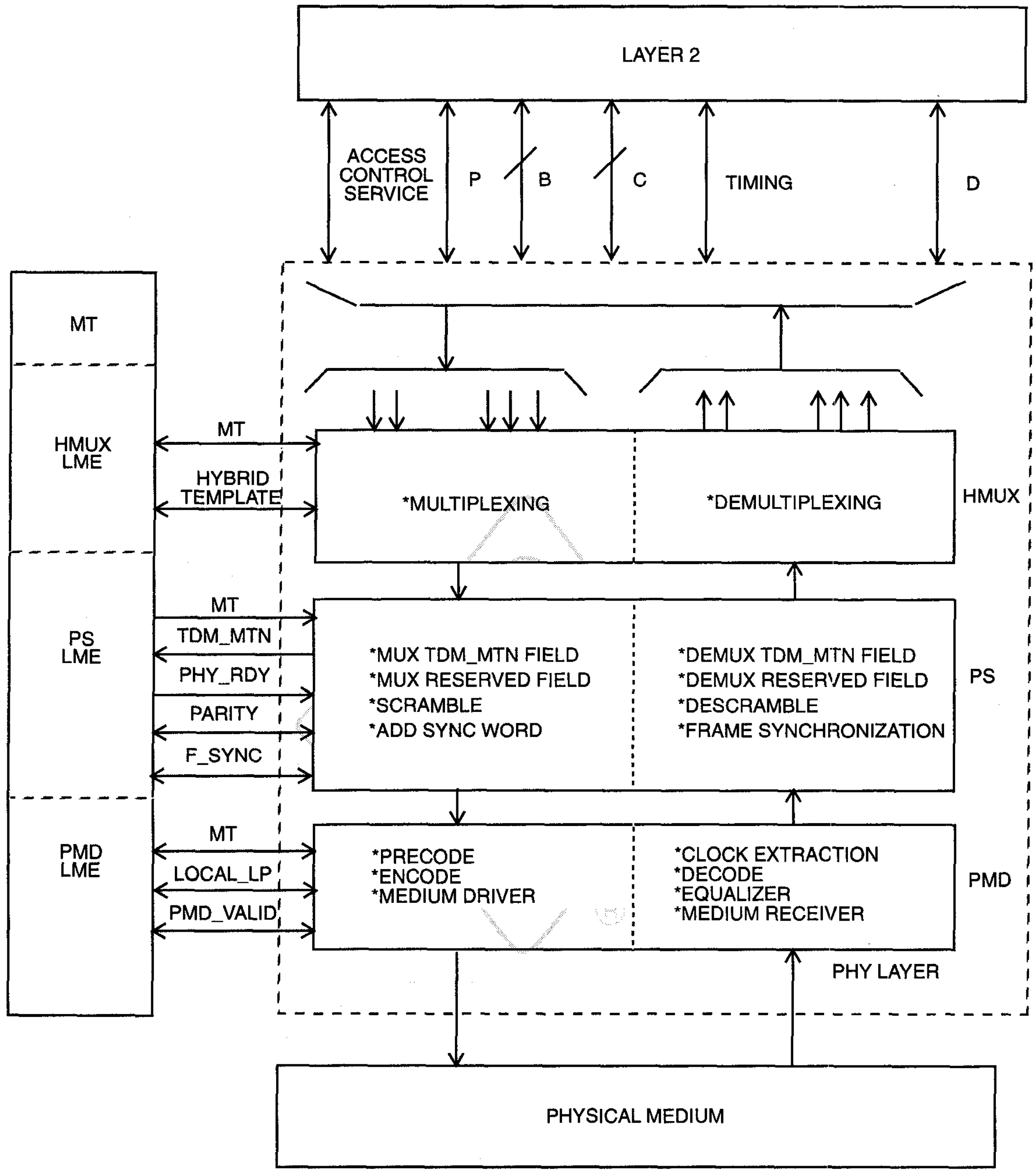


Figure 5-5—IEEE 802.9 ISLAN interface configuration

Chapter 11 and annex D describe the procedures and messages related to the dynamic bandwidth allocation of the 802.9 TDM payload.

5.3.2 PHY Layer

The PHY Layer performs functions such as multiplexing the multiple bit streams, adding and removing TDM framing and synchronization information, and providing access to the transmission medium. As shown in figure 5-3, the PHY Layer is subdivided into the HMUX, physical signalling (PS), and physical medium dependent (PMD) sublayers.

A more detailed model of the PHY Layer is shown in figure 5-5. The functions of each sublayer and the interfaces between the sublayers are described in the following subclauses.

5.3.2.1 HMUX sublayer

The highest sublayer within the PHY Layer is the HMUX sublayer. This sublayer processes the contents of the hybrid multiplexer control (HMC) field in the IEEE 802.9 physical TDM frame. This sublayer provides an interface to the IEEE 802.9 MAC sublayer for the P channel and provides an interface to the user interface of the IEEE 802.9 isochronous services. One of the transmit functions of the HMUX sublayer is to multiplex streams of information from the higher layers and from the HMUX layer management into a single stream of octets for acceptance by the PS sublayer. The HMUX sublayer additionally demultiplexes the stream of octets received by the PS sublayer into a set of channels, each directed to an appropriate higher layer SAP. Finally, the HMUX sublayer provides the functionality needed to dynamically modify the allocation of the PS sublayer's TDM slots to connect and disconnect individual "N × 64 kb/s" channels without disrupting the activity of other channels.

5.3.2.1.1 HMUX/MT service boundary

The interface between the HMUX sublayer and MT includes template synchronization information for a dynamic bandwidth allocation of the P and C channels, generated through the bandwidth management service procedure. The information is mapped and multiplexed onto the HMC field of the TDM frame and transferred to the other peer. The interface also provides for the exchange of HMUX mode parameters.

5.3.2.1.2 HMUX/PS service boundary

The interface between the HMUX sublayer and the PS sublayer conveys the multiplexed information stream composed of P channel, P channel access control information, HMC, D channel, B channels, C channel(s), maintenance information, reserved field information, and timing information.

5.3.2.2 PS sublayer

The PS sublayer is responsible for interfacing between the HMUX sublayer and the PMD sublayer. As such, the PS sublayer is responsible for applying required processing to the multiplexed bit stream and the received bit stream before transmitting them to the PMD sublayer and the HMUX sublayer, respectively. The PS sublayer has an interface to layer management.

As shown in figure 5-5, the PS sublayer is responsible for a variety of functions. In the transmit direction, it completes the formation of an entire TDM frame by appending maintenance information, calculating parity, adding the resulting parity bit, scrambling the bit stream, and appending framing information. In the receive direction, the PS sublayer is responsible for establishing frame synchronization and removing the framing information field. The PS sublayer is responsible for the descrambling function and the parity check function on the received information stream, and for passing maintenance information on to layer management.

5.3.2.2.1 PS/MT service boundary

The interface between the PS sublayer and MT includes the maintenance information, such as activation/deactivation, remote loopback, the bit check result, and the framing status. The framing establishment and other conditions related to receiver function may be used to define PHY READY/NOT READY.

5.3.2.2.2 PS/PMD service boundary

The interface between the PS sublayer and the PMD sublayer is defined in terms of the bit stream forming a complete IEEE 802.9 TDM frame. The service characteristics include quality of service (QOS) parameters. These QOS parameters include but are not limited to:

a) Single bit error (BER) and clock slips (Note that clock slips will be designated in terms of "data invalid." The data invalid signal may be presented 1 or 2 clock times later than the actual occurrence.)

b) Bit error clustering

c) Sublayer delay characteristics

Figure 5-6 illustrates the signal communication between the PS sublayer and the PMD sublayer.

5.3.2.3 PMD sublayer

The PMD sublayer provides the electrical interface to the physical medium. Included under the PMD sublayer are the functions necessary to connect the interface to the medium (UTTP in this case). Specifically, the PMD sublayer defines the medium dependent encoding method, the transmitter and receiver, physical cable and connectors, and other related issues. The PMD sublayer has an interface to layer management.

The IEEE 802.9 PS/PMD service boundary specification supports multiple media dependent sublayers. Therefore, multiple PMD definitions are possible. In this standard, two separate PMD specifications are described supporting two distinct premise distribution loop length configurations. The operational properties of the PMDs describe the properties and functionality for the transmitter and the receiver. This functionality includes a description of the pulse mask, transmitter output power spectrum and impedance characteristics, receiver input impedance characteristics, and the crosstalk and impulse noise requirements. This standard specifies the PMDs for:

— 4.096 Mb/s aggregate rate at 450 m
— 20.480 Mb/s aggregate at 135 m

5.3.2.3.1 PMD/MT service boundary

The interface between the PMD sublayer and MT includes the maintenance information, such as local loopback at the PMD/media interface.

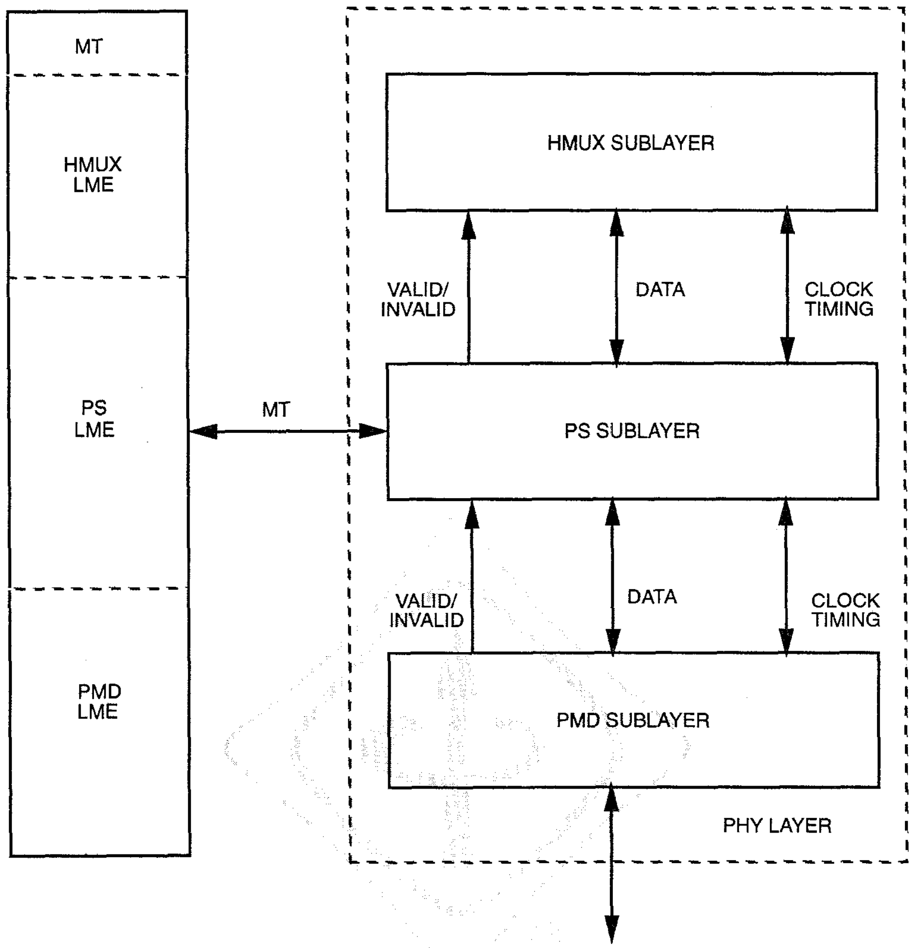


Figure 5-6—PS to PMD service boundary

6. Frame structure

6.1 Overview

This standard provides for two types of frames: the Physical (PHY) Layer's synchronous time division multiplexed (TDM) frame and the Link Layer's medium access control (MAC) frame.

6.1.1 TDM frame

The TDM frame structure is a set of octets which together form the contents of the P, C, B, and D channels along with the synchronization, control, and maintenance information. The frame is generated synchronously every 125 µs. Architecturally, this standard provides for the use of line rates that are a multiple of 4.096 Mb/s, thereby affording the opportunity to effect a reach/rate tradeoff within the standard. In this standard, two rates are specified: 4.096 Mb/s and 20.48 Mb/s. Respectively, these TDM frames contain 64 octets and 320 octets. These specific rates are each supported by their own physical medium dependent (PMD) sublayer specifications which are separately described in Chapter 9.

6.1.2 MAC frame

The MAC frame is delimited from the information carried within the P channel when signalled to do so, or by the access control (AC) channel. These asynchronously delivered MAC frames are separated by padding interframe "idle" octets.

6.2 Numbering convention

The basic convention used for grouping bits is illustrated in figure 6-1. The bits are grouped into octets. The bits of an octet are shown horizontally and are numbered from 0 to 7. Multiple octets are shown vertically and are numbered from 0 to n–1.

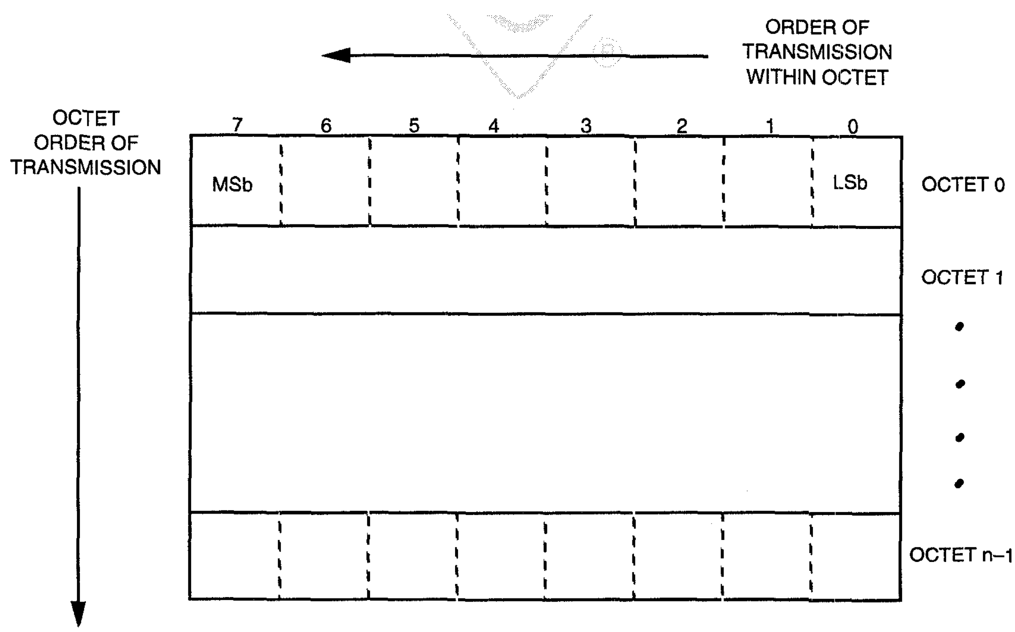


Figure 6-1—Format convention

It should be noted that the CCITT Recommendations use a bit ordering that is numbered from 1 through 8 within an octet.

6.3 Order of bit transmission

In general, within an octet of an ordinary field, bit 0 is the first bit to be transmitted.

6.3.1 Order of bit transmission within FCS field

Within an octet of the FCS field, bit 7 is the first bit to be transmitted.

6.4 Order of octet transmission for fields which span more than one octet

In general, octets are transmitted in ascending numerical order. Octet 0 is the first octet to be transmitted.

6.4.1 Field organization

When a specific field has more than one octet, the transmission of bits shall obey the following field organization.

Assume that a field consists of two octets with the MSb labeled as bit 15 and the LSb labeled as bit 0. The MSB (octet) is organized with the Msb containing bit 15 and the Lsb containing bit 8. Similarly, the LSB is organized with the Msb containing bit 7 and the LSb containing bit 0. The order of transmission shall be:

bits 8, 9, 10, 11, 12, 13, 14, 15, 0, 1, 2, 3, 4, 5, 6, 7

To accomplish this order of transmission, octet 0 will contain bits 15, 14, 13, 12, 11, 10, 9, 8; in the bit positions of 7, 6, 5, 4, 3, 2, 1, 0. Octet 1 will contain bits 7, 6, 5, 4, 3, 2, 1, 0; in the bit positions of 7, 6, 5, 4, 3, 2, 1, 0, respectively.

6.4.2 Order of octet transmission for FCS octets

For the FCS field, the high numbered octet is sent first. Then the remaining octets are sent in descending order.

6.5 TDM frame structure

6.5.1 Overview

The following subclauses overview the frame structure for the IEEE 802.9 TDM frame. Chapter 9 defines the details of these structures and the procedures for the use thereof.

6.5.2 TDM frame format

Figure 6-2 illustrates the TDM frame format. The TDM frame shall have a 125 μs cycle period. Within the frame, octet-based fields are defined. The fields contained in the TDM frame are as follows:

a) Synchronization (SYN)
b) TDM maintenance field (TDM_MTN)
c) Hybrid multiplexer control (HMC)
d) Reserved field
e) D channel (D)

f) B1 channel (B1)
g) B2 channel (B2)
h) Access control (AC)
i) Payload [for carrying information for the C channel(s), and the P channel]

OCTETS

0	1	2	3	4	5	6	7	8 … n–1
SYN	TDM MTN	HMC	RES	D	B1	B2	AC	PAYLOAD

Figure 6-2—TDM frame format

Note that at medium rates above 4.096 Mb/s, it may be necessary to repeat the AC field at a periodic rate of every "n" octets to minimize the buffer requirements at the transceiver and to improve the access response time. The value of "n" is a matter for further study.

6.5.3 Synchronization field

This field is used for the purpose of establishing synchronization between the ISTE and the access unit (AU) with respect to the exchange of the physical TDM frame. This SYN field occupies the first octet of the TDM frame.

The synchronization field is illustrated in figure 6-3.

BITS

7	6 5 4 3 2 1 0
RESERVED	FRAME ALIGNMENT WORD (FAW)

Figure 6-3—TDM frame's synchronization field

The synchronization pattern transmitted by the AU to the ISTE is asymmetric to the pattern transmitted by the ISTE to the AU so that the data stream will avoid misalignment caused by crosstalk noise. The frame alignment word (FAW) transmitted by the AU to the ISTE shall be:

6	5	4	3	2	1	0
0	1	0	0	1	1	1

where the least significant bit of the FAW (bit 0) is transmitted first.

The FAW transmitted by the ISTE to the AU shall be:

6	5	4	3	2	1	0
1	0	1	1	0	0	0

where the least significant bit of the FAW (bit 0) is transmitted first.

Bit 7 is reserved for future standardization use. It shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU. This bit has been reserved for the function of a multiframing bit. In particular, these multiframing procedures are proposed to assist in the ability of the IEEE 802.9 interface to support subrate channels. The use of this bit is a matter for further study. The use of this bit for a proprietary application may lead to incompatibilities with future versions of this standard.

The procedures for the synchronization operation are defined in 9.3.1.

6.5.4 TDM maintenance (TDM_MTN) field

This field is used to transmit local PHY Layer status and control information to the far end of the link. In the PS sublayer's transmitter, maintenance information is received from the layer management entity (LME) and multiplexed into the outgoing bit stream to be transmitted to the far end of the link. In the PS sublayer's receiver, this field is demultiplexed out of the incoming bit stream and presented to the LME.

This TDM_MTN field resides in the second octet of the TDM frame. The field is encoded as shown in figure 6-4.

BITS

7	6	5	4	3	2	1	0
RES	RLBRESP	RLBREQ	LLB	LOCK	PERR	PODD	PEVN

Figure 6-4—TDM frame's TDM_MTN field

The summarized list of functions for the TDM_MTN field is given in table 6-1.

The functions of the bits encoded in this field are defined in 9.3.5. The initialization procedure involving the TDM maintenance field is defined in 9.3.7. The loopback procedures involving the TDM maintenance field are defined in 9.3.9.

6.5.5 Hybrid multiplexer control (HMC) field

The IEEE 802.9 ISLAN interface supports multiple isochronous applications as well as provides an 802-based packet service and other framed services over the P channel. The principal negotiation mechanism for providing configuration management of the isochronous channels within this payload component is through the use of the CCITT Q.93x family of signalling protocols (see Chapter 11 and annex D). However, it is important to note that the reconfiguration of the bandwidth in the TDM frame shall be synchronized between the AU and the ISTE. The HMC field contains the control information that effects the synchronization and determines the D channel rate (16 kb/s or 64 kb/s).

Table 6-1—Functional summary of TDM_MTN field

Name	Function	Value
RES	Reserved	Transmitted as 0 and ignored upon receipt)
RLBRESP	Remote loopback response	0=Deactivated 1=Activated
RLBREQ	Remote loopback request	0=Deactivate request 1=Activate request
LLB	Local loopback status	0=Deactivated 1=Activated
LOCK	Local receiver frame status	0=Unlocked 1=Locked
PERR	Local parity check status	0=Passed 1=Failed
PODD	Cumulative odd bits parity checksum	(x)
PEVN	Cumulative even bits parity checksum	(x)

This HMC field resides in the third octet of the TDM frame. The field is encoded as shown in figure 6-5.

BITS

7	6	5	4	3	2	1	0
DCR	RESERVED		MODE			TXC	HMF

Figure 6-5—TDM frame's HMC field

6.5.5.1 D channel rate select bit (bit 7)

The D channel rate select (DCR) bit is a single control/status bit employed in the HMC field to notify the HMUX sublayer about whether the D channel is to operate in a 16 kb/s (D_{16}) or a 64 kb/s (D_{64}) mode of operation. When the D_{16} option is selected, the HMUX sublayer shall apply the transmitted information in the two least significant bits of the D channel. The six more significant bits of the D channel shall be set to "0" as this bit stream is applied toward the PMD transmitter.

The bit encoding is described below.

Bits 7	
0	D_{16} operation
1	D_{64} operation

6.5.5.2 Reserved bits (bits 6 and 5)

These bits are reserved for future standardization use. Since the entire HMC field is considered to be delivered to the peer entity as a part of the service agreement of the HMUX sublayer to its LME and its adjoining Layer 2 uses, the contents of this field shall be delivered to the peer entity. These bits shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU. The use of these bits for a proprietary application may lead to incompatibilities with future versions of this standard.

6.5.5.3 Template identification MODE bits (bits 4, 3, and 2)

Template identification MODE bits are used to designate a bandwidth allocation format or "mode" by which the ISTE and AU communicate for the allocation of bearer service across the IEEE 802.9 interface. The modes describe degrees of flexibility with respect to how bandwidth is allocated. These modes vary from a single service (IEEE 802 only) to a more involved fully dynamic CCITT Q.93x message-based approach for bandwidth management.

The purpose of the modes is to allow lower functionality equipment to enjoy the utilization of the channel structures without having to engage in a fully dynamic network management scheme.

Each mode of operation can be described as a template of the channel configuration of the IEEE 802.9 TDM frame operation. The location of the C channel(s) isochronous bandwidth effectively determines the P channel bandwidth since the unallocated isochronous bandwidth may revert to the P channel MAC transport.

A more complete functional description of the modes is provided in annex E.

The bit encoding is described below.

Bits 4 3 2	
0 0 0	Mode 0 - IEEE 802 bearer service only
0 0 1	Mode 1 - ISDN basic rate bearer service only
0 1 0	Mode 2 - IEEE 802 bearer service and ISDN BRI service only
0 1 1	Mode 3 - Multiservice dynamic bandwidth management that is controlled by CCITT Q.93x
1 0 0	Reserved for future IEEE 802.9 standardization
1 0 1	-
1 1 0	-
1 1 1	Reserved for future IEEE 802.9 standardization

Mode 0 is a template that describes IEEE 802 bearer service only. In this mode, the terminal equipment has a need to use only the "802 LAN" component of the bandwidth. Thus, only the P channel portion of the bandwidth is actively used. The need for the support of the ISDN CCITT Q.93x based signalling shall be limited to the minimum functionality as defined in the local bandwidth management procedures. These management procedures are a matter for further study.

Mode 1 is a template that describes ISDN basic rate service only. In this mode, the terminal equipment uses only the "ISDN Basic Rate" component of the bandwidth. Thus, only the fixed D channel, B1 channel, and B2 channel portion of the bandwidth are actively used. The P channel and C channels are not available for use. CCITT Q.93x based signalling is essential for the negotiation of call setup for the establishment of circuit-switched and packet-switched channels.

Mode 2 is a template that describes an interface which supports IEEE 802 bearer services and ISDN basic rate bearer services only. This may be thought of as a concatenation of Mode 0 and Mode 1 capabilities.

Mode 3 is a template that describes a fully dynamic service where the bandwidth of the interface is governed by CCITT Q.93x. This is an environment where the complete set of ISDN and IEEE 802 LAN services may be simultaneously supported.

A detailed description of the interoperability of terminal equipment that supports different modes is provided in 9.2.3.1.

6.5.5.4 Template exchange notification bit (bit 1)

The template exchange notification (TXC) bit is a single status bit employed in the HMC field to notify the peer HMC entity that it has effected a reconfiguration of channel bandwidth assignment as specified by the CCITT Q.93x management/signalling messages from the AU.

In a minimum configuration, the ISTE and AU may alternatively specify an identifier (MODE) which specifies a template. This template is essentially a map that describes the specific aggregate channel configuration. The procedures for TXC are defined in 9.17.

The bit encoding is described below.

Bit 1	
0	Template exchange (reconfiguration) not complete
1	Template exchange complete

6.5.5.5 HMUX multiframing bit (bit 0)

The HMUX multiframing (HMF) bit is used to guard template exchange timing integrity. The HMF bit is asserted in the HMC field of every eighth TDM frame. It provides a multiframing pattern by which peer HMUX sublayers can synchronize their actions. The procedures for this multiframe operation are defined in 9.14 through 9.16.

6.5.6 Reserved field

This field resides in the fourth octet of the TDM frame.

These bits are reserved for future standardization use. Since the entire reserved field is considered to be delivered as part of the service agreement of the HMUX sublayer to its LME and its adjoining Layer 2 uses, the contents of this field shall be delivered to the peer entity. These bits shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU. The use of these bits for a proprietary application may lead to incompatibilities with future versions of the IEEE 802.9 ISLAN standard.

6.5.7 D channel field

This field resides in the fifth octet of the TDM frame.

The IEEE 802.9 D channel provides for both a 16 kb/s basic rate interface (BRI) and also for a 64 kb/s primary rate interface (PRI) ISDN bearer service. It is not ISDN conformant at the PHY Layer (CCITT I.430) but it does provide, at the IEEE 802.9 PHY Layer isochronous service interface, the capability of delivering ISDN-conformant higher layer (Layer 2 and above) services over the IEEE 802.9 interface. The use of the D channel may be restricted to conveying signalling information only in some applications. In this situation, the ISDN CCITT Q.93x message set is extended to support the establishment (disestablishment) of IEEE 802.9's additional isochronous services. All information in the channel is packetized using the CCITT Q.93x family of protocols.

For the 16 kb/s configuration, the D channel information shall be transmitted across bit positions 0 and 1 of octet 4 of the TDM frame. Bits 2 through 7 shall be set to "0" by the transmitter and ignored by the receiver for 16 kb/s operation.

6.5.8 B1 channel field

This field resides in the sixth octet of the TDM frame.

The B1 channel provides a 64 kb/s, duplex isochronous access. This channel may support all ISDN bearer services. The information on the B1 channel may be nonswitched or either circuit or packet switched depending on user request and network capabilities.

6.5.9 B2 channel field

This field resides in the seventh octet of the TDM frame.

The B2 channel provides a 64 kb/s, duplex isochronous access. This channel may support all ISDN bearer services. The information on the B2 channel may be nonswitched or either circuit or packet switched depending on user request and network capabilities.

6.5.10 Access control (AC) field

The AC field carries information relating to the MAC for the request/grant arbitration procedure used in governing access to the P channel for MPDU transport.

The AC field resides in the eighth octet of the TDM frame. The field is encoded as illustrated in figure 6-6.

BITS

7	6	5	4	3	2	1	0
RESERVED		SECURE	REQ3	REQ2	REQ1	GRANT	SOF

Figure 6-6—TDM frame's access control field

6.5.10.1 Reserved bits (bits 7 and 6)

These bits are reserved for future use within the IEEE 802.9 ISLAN standard. Since the AC field is considered to be delivered to the peer entity as a service, the contents of this field shall be delivered to the peer entity. These bits shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU. The use of these bits for a proprietary application may lead to incompatibilities with future versions of this standard.

6.5.10.2 Secure information bit (bit 5)

This field is reserved for use as a secure exchange PHY control signal for the establishment of isochronous services between the AU and the ISDN WAN. The procedures for a security exchange mechanism at the PHY sublayer are a matter for further study. It has been proposed that the secure access exchange utilize a CCITT Q.93x management message as derived from the Codeset 7 message set. Since the details of this feature remain under investigation, this physical exchange bit has been reserved as an alternative communication transport method. Annex G describes an optional set of procedures for an ISTE security management entity to access a secure control device (SCD) via the exchange of information in this field.

The bit encoding is described below.

Bits 5	
0	No security
1	Invoke security

6.5.10.3 Request bits (bits 4, 3, and 2)

The request bits (REQ3, REQ2, and REQ1) are used to provide an indication of physical access priority as the ISTE requests access of the shared resources in the AU. Note that although the IEEE 802.9 topology is modeled as a direct star-wired connection from each ISTE to the AU, the AU may support many ISTEs, which are each sharing a finite set of resources in the AU. A range of AU designs and capabilities can be implemented supporting IEEE 802.9's service model and interface architecture. Individual AU designs will involve different degrees of either shared processing, shared memory, or bus bandwidth. Concurrent support for multiple ISTEs requires that the AU implement some arbitration process to manage the use of the available AU's shared resources. Furthermore, support for IEEE 802's multiple priority levels requires that fair mediation among multiple ISTE requests for access to the AU's resources be employed. The request bits are, therefore, used to request access to the shared AU resources with these different priorities.

Figure 6-7 illustrates the ISTE application management scenario wherein two or more types of applications (co-resident on the ISTE) are targeting the P channel for use. The application management entity has the ability to direct that applications 1, 2, and 3 are able to prioritize its request into a low, medium, and high request priority. These three levels of priority will be translated by the IEEE 802.9 MAC management into the request bits of the AC field.

With reference to figure 6-7, the frame control (FC) priority is a part of the MAC and is a separate mechanism from the AC priority. The AC priority deals with physical access of service in the AU, while the FC priority allows for interworking with the network-side LAN. Refer to 6.6.5 for a more detailed description of the FC field.

Note that request priority is needed for communication that is to be sent from the ISTE toward the AU. It is not needed in the direction of communication from the AU toward the ISTE.

The bit encoding is described below.

Bits 4 3 2	
0 0 0	No request
0 1 1	Request priority #1 (Low)
1 0 1	Request priority #2 (Medium)
1 1 0	Request priority #3 (High)

Any other value is considered an invalid request.

In a multiported AU design, the AU shall support three priority buffers (low, medium, and high) for each port that is connected to the AU in order to support the AC priority mechanism.

Refer to 8.3 for a more detailed description of the procedures involved with the exchange of these bits.

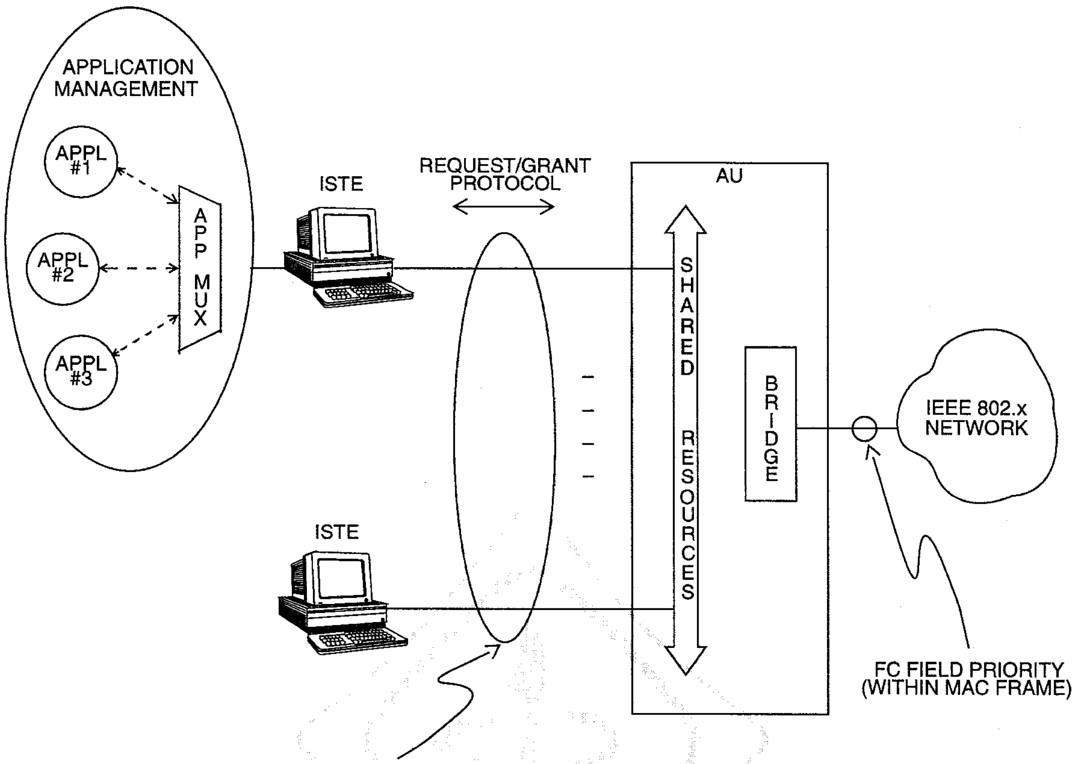


Figure 6-7—Example of AC field priority

6.5.10.4 FC priority to AC priority mapping

The application management entity in the ISTE is presented with two priority mechanisms—AC priority and FC priority—to manage as it effects a quality of service across the IEEE 802.9 subnetwork. Figure 6-8 illustrates the potential combinations of mapping of FC priority into AC priority as the LLC PDU is presented with an FC priority for LLC-to-MAC mapping and an AC priority for MAC-to-PHY mapping.

The most general situation allows for the ability of the application entity to assign the FC priority independently of the AC priority per the following algorithm:

$$AC_{priority} = f_{layer\text{-}mgmt}\ (FC_{priority})$$

This form of flexibility is excessive. Therefore, the FC priority to AC priority association mapping in figure 6-9 is recommended for layer management provisioning.

The ISTE management entity can prioritize these packet-based applications with several levels of priority. Within each of these communications-oriented applications, there may also be layer PDUs with different levels of priority. When one or more of these applications, as illustrated in figure 6-7, uses the IEEE 802.9 MAC to convey an ISO/IEC 10039 MAC frame, the LLC will prioritize the multiple application service queues and offer them to the IEEE 802.9 and LPDU with an appropriate level of priority.

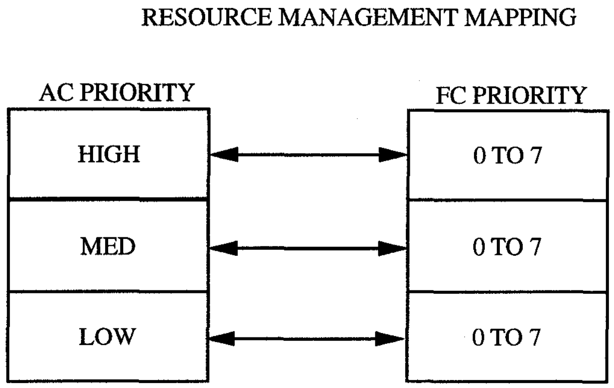


Figure 6-8—Priority resource management mapping combinations

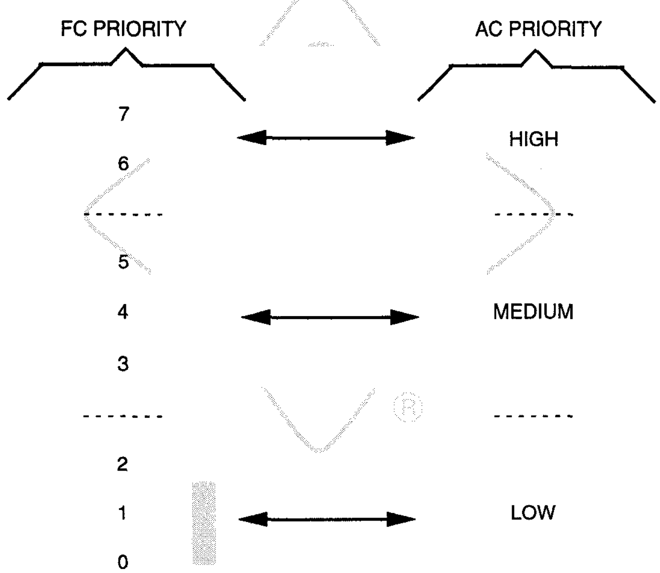


Figure 6-9—Recommended layer management mapping of FC priority into AC priority

When the multiple applications use different Layer 2 protocols on top of the IEEE 802.9 MAC, the service identifier (SID) provides a means of service discrimination. If the ISTE Layer 2 request is for an ISO/IEC 10039:1991-conformant MAC service, then the IEEE 802.9 MAC will invoke the AC request/grant protocol with a high, medium, or low priority encoding that is mapped onto the Layer 2 request. The value that will be inserted into the subsequently transmitted IEEE 802.9 MAC frame's FC field will be as shown in figure 6-9. In the case of support of multiple Layer 2 packet services (e.g., ISO/IEC 10039 and LAPD frames), it is a matter for further study as to how or whether future extensions to this standard will provide MAC arbitration of multiple concurrent service requests. The encodings of the AC field request bits, the SID, and any other FC field parameters to satisfy this type of concurrent service request are also matters for further study.

6.5.10.5 Grant bit (bit 1)

This bit is transmitted as a logic "1" when a P channel packet can be received, and as a logic "0" otherwise.

The bit encoding is described below.

Bit 1	
0	No grant
1	Grant

Refer to 8.3 for a more detailed description of the procedures involved with the exchange of this bit.

6.5.10.6 Start-of-frame bit (bit 0)

The start-of-frame (SOF) bit is a single status bit employed in the MAC frame delimiting operation which notifies the receiver that this INFO field of the MAC frame is the beginning portion of an MPDU.

This bit indicates the starting point of a new MAC frame. This bit is transmitted as a logic "1" when the first bit of a MAC frame is transmitted in the following P channel payload field in that TDM frame, and as a logic "0" otherwise. The first bit of a MAC frame is transmitted from the beginning of the P channel payload field.

While the transport of the SOF bit is supported by the Physical transport, the procedures for the support of this procedure are contained within the MAC procedures. The SOF bit is generated and processed entirely within the MAC.

The bit encoding is described below.

Bit 0	
0	Not start of a MAC frame
1	Start of a MAC frame

This bit is set to logic "1" if this MAC payload unit is the first of a MAC frame. If this MAC payload is a consecutive segment of the current MAC frame being processed, then this bit is set to logic "0." Figure 6-10 illustrates the situation in which a MAC frame (that is awaiting transmission) needs to be segmented across multiple MAC payload units (or alternatively across multiple TDM frames). In this example, the MAC frame will be sent out over three successive TDM frames.

The SOF bit is also associated with the request/grant protocol for provision of MAC service. It provides a timing relationship between request (R) and grant (G). Thus the request/grant protocol is robust with respect to both flow control and resource allocation.

Refer to 8.3 for a more detailed description of the procedures involved with the exchange of this bit.

6.5.11 Payload field

The payload field is composed of two subfields: the service identification (SID) field and the payload information field. Figure 6-11 illustrates the payload field.

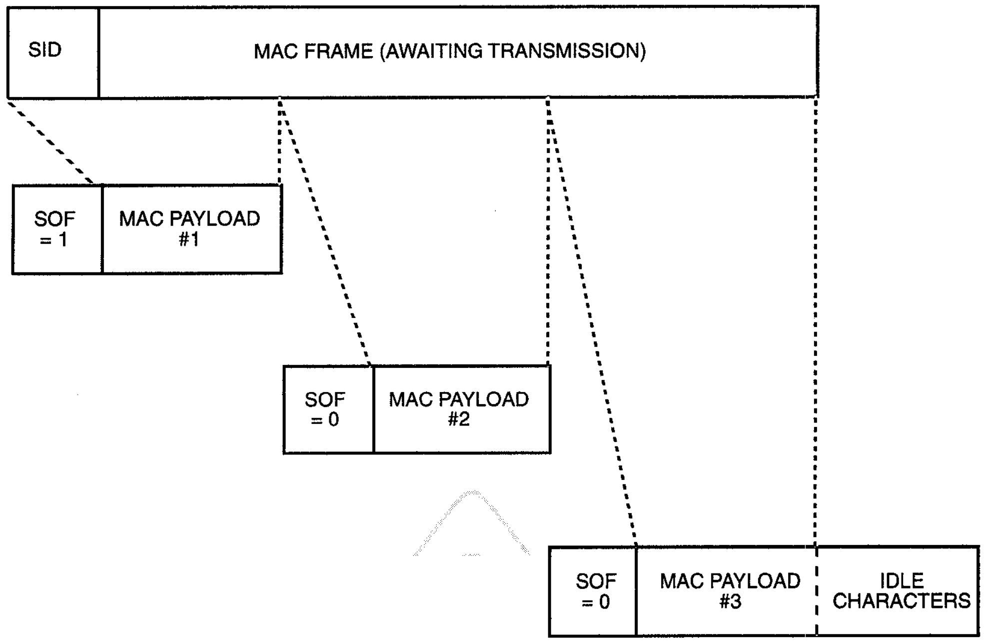


Figure 6-10—MAC frame segmentation across multiple TDM frames

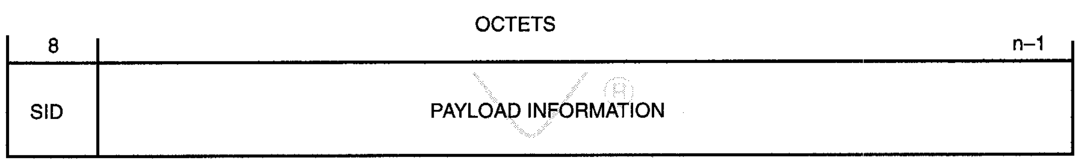


Figure 6-11—TDM payload field

6.5.11.1 SID field

This field exists at the beginning of the packet payload component. It is used to designate the type of service carried within the packet portion of the payload information field. A more complete description of how this field is used is provided in 6.6.2.

6.5.11.2 Payload information field

This field contains the information octets of any additional C channels, and the P channel which may be configured within the available bandwidth of the payload field. Note that the minimum size of the P channel shall be limited by the application requirements of the topology that represents the connection of the AU to the backbone LAN application. The support of station management communication will be an important factor in the characterization of P channel bandwidth size. There shall be only a single P channel allocated within this payload field. There may be multiple B or C channels allocated within the payload field. From a signalling perspective, successive allocations of B channels beyond B_1 and B_2 will be referred to as instances of C_1 channels.

The procedures for the allocation of this bandwidth are described in Chapter 11.

With the 20.48 Mb/s aggregate rate interface, the payload field will be distributed across five rows. This enables more efficient use of payload field bandwidth for the situation in which one MAC frame has terminated and a second MAC frame is queued for transmission. Figure 6-12 illustrates the TDM frame structure for the 20.48 Mb/s interface.

0	1	2	3	4	5	6	
SYN	TDM MTN	HMC	RES	D	B1	B2	
AC	8	63 OCTETS OF PACKET PAYLOAD SPACE					70
AC	72	63 OCTETS OF PACKET PAYLOAD SPACE					134
AC	136	63 OCTETS OF PACKET PAYLOAD SPACE					198
AC	200	63 OCTETS OF PACKET PAYLOAD SPACE					262
AC	264	56 OCTETS OF PACKET PAYLOAD SPACE					319

Figure 6-12—Example of 20.48 Mb/s TDM frame structure

6.6 Packet frame structure

6.6.1 Overview

The following subclauses define in detail the frame structure for the IS systems using the IEEE 802.9 MAC. It defines the relative positions of the various components of the MAC frame and defines a method for representing station addresses. It also defines a method for distinguishing the various LPDUs that the MAC sublayer may carry.

6.6.2 SID field

This field (located at octet 0 of the payload frame) designates the type of service provided within the payload information field. It specifies the type of logical link control (LLC) PDU being carried in the MAC SDU. The MAC convergence function (see 7.3.2.3) uses this information as a service selector to determine where to route the information. The following service types have been defined:

— IEEE 802.2 LLC PDU
— ISDN LAPD (CCITT Q.921) PDU

This ISDN LAPD PDU is reserved for the application of a P_D channel. The use of this method for sending ISDN D channel information facilitates the architecture wherein the D channel information is passed transparently within the P channel across a minimum functionality AU. This is an implementation option.

The bit encoding is described below.

Bits 7 6 5 4 3 2 1 0	
0 1 0 1 0 1 0 1	IEEE 802.2 LLC
0 1 0 1 0 1 1 0	ISDN LAPD

All other values are reserved for future IEEE 802.9 standardization use.

The encoding of the SID field is designed to provide protection against received bit errors. The bit encoding of the field is based on encoding 16 code points as a binary value in the odd bits of the field and the bit inversion of the binary value interleaved in the even bit position within the field. This is graphically illustrated below.

SID field bit 7 6 5 4 3 2 1 0

SID encoding $a\ \bar{a}\ b\ \bar{b}\ c\ \bar{c}\ d\ \bar{d}$

where abcd is the four-bit SID value

If an invalid SID value is received, namely one in which all four odd numbered bit values are not an exact inversion of the four even numbered bit values, a layer management action LM-SID-ERROR is invoked and the MAC convergence function ignores subsequent received information until a valid SOF and SID sequence are subsequently detected.

6.6.3 ISLAN MAC frame format

Figure 6-14 shows the fields of the MAC frame. It consists of the following:

a) The length field = 2 octets
b) The frame control (FC) = 1 octet
c) The destination address (DA) = 6 octets
d) The source address (SA) = 6 octets
e) The MAC information (contains the LLC PDU)
f) The frame check sequence (FCS) = 4 octets

The entire MAC frame (consisting of length, FC, DA, SA, information, and FCS fields) shall not exceed 5119 octets. All of the fields are of fixed sizes except for the MAC information field, which may contain an integer number of octets not exceeding a maximum value of 5100 octets. The minimum size of the MAC frame shall be 19 octets.

Relative to figure 6-13, the octets of the MAC frame are transmitted from left to right.

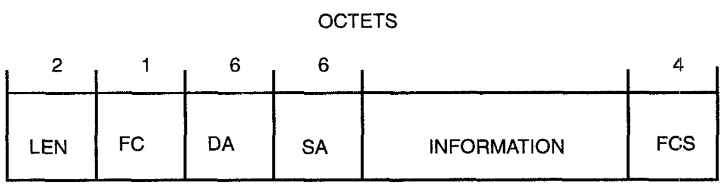


Figure 6-13—MAC frame format

6.6.4 Length field

The length field is located in octets 1 and 2 of the MAC frame, and indicates the total octet count in the MAC frame, excluding the SID, length, and FCS fields. The starting point of the MAC frame is provided in the AC field of the TDM frame as described in 6.5.2; the end of the MAC frame is calculated using the length field information.

The format of the length field is illustrated in figure 6-14. The most significant three bits (bits 15, 14, and 13) are reserved. These bits shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU. The use of these bits for a proprietary application may lead to incompatibilities with future versions of this standard.

BITS

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	LENGTH												

Figure 6-14—MAC length field

6.6.5 FC field

The FC field (located at octet 3 of the MAC frame) designates the priority of the MAC frame. This priority is used by the AU when it maps a MAC frame from the IEEE 802.9 subnetwork over the LAN backbone network. Refer to 6.5.10.4 for details of the relationship between the FC field encoding and the AC field encoding for the request priority.

The bit encoding is described below.

Bits 2 1 0	
0 0 0	Lowest priority request
0 0 1	-
0 1 0	-
0 1 1	-
1 0 0	-
1 0 1	-
1 1 0	-
1 1 1	Highest priority request

Bits 3 through 7 are reserved for future standardization use. These bits shall be set to logic "0" by the transmitter.

6.6.6 Address fields

Each MAC frame shall contain two address fields: the destination address field and the source address field, in that order. The destination address field shall specify the destination addressee(s) for which the frame is intended. The source address field shall identify the station from which the frame is initiated.

The representation of each address field shall be as follows:

— Each address field shall contain 48 bits.

— The first bit delivered to/received from the MAC sublayer—the least significant bit (LSb)—shall be used in the destination address field as an address type designation bit to identify the destination address either as an individual or a group address. In the source address field, the first bit is reserved and set to "0." If this bit is "0," it shall indicate that the address field contains an individual address. If this bit is "1," it shall indicate that the address field contains a group address that identifies none, one or more, or all of the stations connected to the ISLAN.

— The second bit delivered to/received from the MAC sublayer—the second LSb—shall be used to distinguish between locally or globally administered addresses. For globally administered (or U for universal) addresses, the bit is set to "0." If an address is assigned locally, this bit shall be set to "1." Note that for broadcast address, this bit is also set to "1."

— Each octet of the address field is transmitted LSb first.

A MAC sublayer address is one of the following two types:

a) *Individual address.* The address associated with a particular station on the network.

b) *Group address.* A multidestination address, associated with one or more stations on a given network. There are two kinds of multicast address:

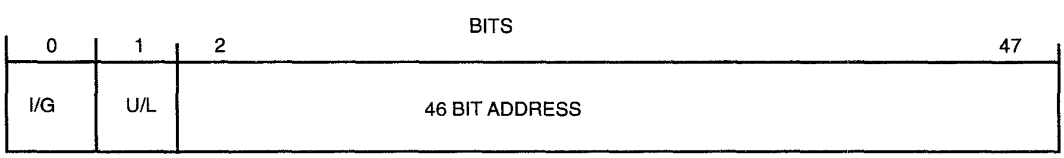
   1) *Multicast-group address.* An address associated by higher-level convention with a group of logically related stations.

   2) *Broadcast address.* A distinguished, predefined multicast address that always denotes the set of all stations on a given network.

   All "1's" in the destination address field shall be predefined to be the broadcast address. This group shall be predefined for each network to consist of all stations actively connected to that network. All stations shall be able to recognize the broadcast address.

The address space shall also be partitioned into locally administered and globally administered addresses. The details of this procedure are beyond the scope of this standard but may be available in other IEEE 802 documents.

Figure 6-15 illustrates the address field format for IEEE 802 styled MAC frames.



I/G = 0 INDIVIDUAL ADDRESS
I/G = 1 GROUP ADDRESS

U/L = 0 GLOBALLY ADMINISTERED ADDRESS
U/L = 1 LOCALLY ADMINISTERED ADRESS

Figure 6-15—Address field format

6.6.6.1 Destination address field

The destination address field specifies the station(s) for which the frame is intended. It may be an individual or multicast (including broadcast) address.

6.6.6.2 Source address field

The source address field specifies the station sending the frame.

6.6.7 MAC information field

The maximum size of a MAC information field shall be 5100 octets. The minimum size of an individual MAC information field shall be 0 octets. The length of this field is an integer number of octets. If the higher layer data unit to be transmitted does not consist of an integer multiple of eight bits, the final octet shall be stuffed with "0" bits to form a complete octet and maintain octet boundary alignment.

6.6.8 Frame check sequence field

The MSDU field contains a sequence of "n" octets. Full transparency is provided in the sense that any arbitrary sequence of octet values may appear in the data field up to a maximum to be specified. A cyclic redundancy check (CRC) is used by the transmit and receive algorithms to generate a CRC value for the FCS field. The FCS field contains a 4-octet (32-bit) CRC value. This value is computed as a function of the contents of the LEN, FC, DA, SA, and information fields. The encoding is defined by the following generating polynomial.

$$G(x)=x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

Mathematically, the CRC value corresponding to a given frame is defined by the following procedure:

a) The first 32 bits of the frame are complemented (perform a logical addition with an all "1's" pattern).

b) The "n" bits of the frame are then considered to be coefficients of a polynomial M(x) of degree "n–1." The first bit of the SID field corresponds to the $x^{(n-1)}$ and the last bit of the LPDU field corresponds to the x^0 term.

c) M(x) is multiplied by x^{32} and divided by G(x), producing a remainder R(x) of degree less than 31.

d) The coefficients of R(x) are considered to be a 32-bit sequence.

e) The bit sequence is complemented and the result is the CRC.

The 32 bits of the CRC value are placed in the frame check sequence field so that the x^{31} term is the leftmost bit of the higher order octet, and the x^0 term is the rightmost bit of the lower order octet. The bits of the CRC are thus transmitted in the order x^{31}, x^{30},..., x^1, x^0.

6.6.9 P channel idle pattern

When there is no information to transmit, the sending side shall transmit the idle pattern consisting of all "1" bits; the receiving side shall ignore this information.

7. Service specification

This chapter describes the services provided at the IEEE 802.9 medium access control (MAC) and Physical (PHY) interface. It encompasses services provided on the packet channels (i.e., the P and D channels), services provided on the isochronous bearer channels (i.e., the multiple B and C channels), and management.

7.1 Packet services

Packet services consist of user information as well as signalling information.

There are two methods for transporting signalling information across an IEEE 802.9 interface. These lead to different requirements on the service provided by the MAC. The two methods are as follows:

a) *Signalling over the D channel.* This information is provided in a channel other than the one over which the MAC is defined. In this case the support of services other than LLC will be dictated by whether or not the P channel will be used as a transmission channel for the transport of ISDN bearer service. In this application, the support of the ISDN bearer service will, very likely, imply the support of LAPD.

b) *Signalling over the P_D channel.* This information is provided within the channel over which the MAC is defined. Here the support of LAPD by the MAC becomes mandatory. In this case, the support of ISDN bearer services, which depend on LAPD at Layer 2, is accommodated by the transport of this signalling information. The requirements for transport of ISDN signalling information over the P_D channel are a matter for further study.

7.1.1 LLC/MAC service

The IEEE 802.9 MAC provides services to support data transport between peer MAC service users at the Logical Link Control (LLC) level. The IEEE 802.9 MAC provides services to support the IEEE 802 data services and optionally, the signalling used in the provision of ISDN service (P_D channel). The IEEE data consists of the LLC service data units (SDUs); the signalling information uses LAPD protocol data units (LLC Type 2). The support of other packet-mode bearer services is a matter for further study.

7.1.2 Detailed specification of the MAC service primitives

ISO/IEC 10039:1991 defines the connectionless service provided by the MAC sublayer entity to peer LLC entities to permit the exchange of LLC protocol data units (LPDUs), through the use of MAC service data units (MSDUs). The following description provides a summary of the MAC service primitives; full details are provided in ISO/IEC 10039:1991.

In an ISLAN, this service is used for the exchange of LPDUs, each containing an OSI Network Layer PDU. In addition, an MSDU may contain an LPDU, encapsulating a Layer 3 PDU of the CCITT Q.93x family of signalling protocols. Finally, the MSDU may contain an LAPD PDU encapsulating a CCITT Q.93x or other PDU. In order to accomplish the delivery of both LPDUs and LAPD PDUs using the MAC service, a MAC convergence function is defined (see 7.2.3).

The service primitives of the MAC service are as follows:

a) MA-UNITDATA request
b) MA-UNITDATA indication

All of the information required for the exchange of LPDUs, submitted by the MAC service user as MSDUs, is contained within the appropriate service primitive and its parameters. Therefore, no connection setup is required prior to the exchange of LPDUs.

Each service interaction takes the form of the passing of a service primitive between the MAC service user and the MAC service provider. The MAC service provider considers each service interaction to be completely independent from all other such service interactions.

There is no defined flow control service provided across the MAC service boundary by the service provider.

Abstract Syntax Notation 1 (ASN.1) is used below in the specification of the semantics of the service primitives as a convenient, precise syntax. This notation is also used elsewhere in the ISLAN service specification for the definition of the service boundaries presented to the layer management entities (LMEs). However, it should be stressed that no use of the ASN.1 Basic Encoding Rules is intended or implied.

NOTE—Some of the features of the ASN.1 descriptive language used in this chapter are as follows:

a) The association operator "::=" is interpreted as "is defined as."
b) The structure "CHOICE" is a list of members for which one and only one member is to be selected for use. The members are defined in terms of both a textual name, a "tag" field that is expressed in square brackets, and a description of the ASN.1 data type. The tag field is expressed as [x] where "x" is the value that would be used by a common management information protocol (CMIP) management process.
c) The structure "ENUMERATED" is an ordered list of scalar constants. The member reference is via a combination of abbreviated name and value which is expressed in parentheses, e.g., "act(0)." All members of an enumerated list are of the same data type. This data type is "integer."
d) The structure "SET" is an unordered list of specified data types.
e) The structure "SEQUENCE" is an ordered list of specified data types.

7.1.3 MA-UNITDATA request

This primitive defines the transfer of an MSDU from a local LLC sublayer entity to its MAC service provider.

Semantics of the service provider

The semantics of the primitive are as follows:

```
MA-UNITDATA request::= SEQUENCE    {
                                     da    [0]   Destination-Address
                                     sa    [1]   Source-Address
                                     data  [2]   MSDU
                                     pri   [3]   Priority
                                    }
MSAP-Id::= OCTET STRING (SIZE(6))
Destination-Address::= MSAP-Id
Source-Address::= MSAP-Id
MSDU::= OCTET STRING
Priority::= INTEGER
```

Each of the destination address (of the remote MSAP) and the source address (of the local MSAP) parameters has, as a value, a 48-bit MSAP identifier (or MAC address). This standard does not support the 16-bit MSAP identifier.

The destination address may be either an individual or a group address; the source address is always an individual address. The group address will be either a broadcast address for all MSAPs in a single LAN or a multicast address for a subset of the MSAPs in a LAN. Both the destination address and the source address may be either universally or locally administered.

The MSDU parameter has, as a value, an octet string which is the LPDU to be transferred to a peer MAC service user. It shall always contain an integer number of octets within the range bounded by the minimum and maximum LPDU sizes.

The priority parameter has, as a value, a requested priority for the MPDU transfer. The value may be any integer value defined as meaningful for the MAC service provider.

When generated

This primitive is generated by the LLC sublayer user of the MAC service whenever it has an MSDU to submit for the purpose of transferring an LPDU to a remote peer user of the MAC service. This MSDU may be generated in response to a request from higher layers of protocol or from events that are internal to the LLC sublayer, such as responses required by Type 2 operations, as defined by ISO/IEC 8802-2:1994.

Effect of receipt

The receipt of this primitive by the MAC service provider shall cause the MAC entity to prefix or append all MAC specific fields to the MSDU in order to create an MPDU, which will be submitted by the MAC entity to the PHY Layer for transmission as a stream of symbols. (See the PHY service specification below.)

7.1.4 MA-UNITDATA indication

This primitive defines the transfer of an MSDU from a MAC service provider to its local LLC sublayer entity.

Semantics of the service primitive

The semantics of the primitive are as follows:

```
MA-UNITDATA indication::= SEQUENCE  {
                                      da    [0]   Destination address
                                      sa    [1]   Source address
                                      data  [2]   MSDU
                                      pri   [3]   Priority
                                     }
MSAP-Id::= OCTET STRING (SIZE(6))
Destination-Address::= MSAP-Id
Source-Address::= MSAP-Id
MSDU::= OCTET STRING
Priority::= INTEGER
```

Each of the destination address (of the local MSAP) and the source address (of the remote MSAP) parameters has, as a value, a 48-bit MSAP identifier (or MAC address). This standard does not support the 16-bit MSAP identifier.

The destination address may be either an individual or a group address; the source address is always an individual address. The group address will be either a broadcast address for all MSAPs in a single LAN or a multicast address for a subset of the MSAPs in a LAN. Both the destination address and the source address may be either universally or locally administered.

The MSDU parameter has, as a value, the LPDU which has been transferred from a remote peer MAC service user. It shall always contain an integer number of octets within the range bounded by the minimum and maximum LPDU sizes.

The priority parameter has, as a value, the priority for the delivered MPDU. The priority specified in the primitive is the value requested at the remote MSAP, unless this was modified by the MAC service provider, in which case, the parameter has the modified value.

When generated

This primitive shall be generated by the MAC service provider and passed to the LLC entity whenever the service provider has an MSDU to deliver to its local user for the purpose of delivering an LPDU from a remote peer user of the MAC service. This primitive shall be passed to the MAC service user only if the MPDU received has been determined by the service provider to be uncorrupted and only if the value of the destination address parameter is identical to the identifier of the MSAP across which the MA-UNITDATA indication primitive is to be delivered.

Effect of receipt

The effect of the receipt of this primitive by a MAC service user is determined by the content of the MSDU, as interpreted by the LLC protocol functions.

7.1.5 PHY service

7.1.5.1 Introduction

This subclause defines an abstract specification of the PHY service which is independent of the differentiating characteristics of the variety of PHY service users. It therefore specifies one set of services provided both to users that require an asynchronous, packet-mode transmission service, and to users that require a synchronous circuit-mode transmission service.

This PHY service specification is an extension to the OSI PHY Layer service, because of the requirement to provide a timing service to/from the PHY service provider from/to the PHY service user [and a timing service in the opposite direction at the access unit (AU) when it is synchronized with the synchronous digital hierarchy of a public/private network]. Timing is required by those PHY service users which are sources/sinks of information sampled at uniform intervals for transmission in an isochronous channel (e.g., digital voice applications).

While defining a PHY service that is suitable both for users of an asynchronous service and for users requiring network timing extraction, the effort has been made to remain as consistent as possible with ISO/IEC 10022:1990.

7.1.5.2 Overview of the service

The PHY service provides a single set of PHY Layer services to all sources/sinks of ISLAN information flows, e.g., the ISLAN MAC, a LAPD entity, other Data Link (DL) protocol entities (e.g. a LAPB entity), and sources/sinks of information (e.g., voice and video) sampled at uniform intervals for transmission in an isochronous channel.

The physical medium dependent (PMD) sublayer provides the electrical interface to the physical medium for the purpose of conveying contradirectional continuous bit streams between the ISTE and the AU. The physical signalling (PS) sublayer above the PMD forms a cyclic series of octet wide slots to constitute a time division multiplexed (TDM) frame of 125 μs duration. It is then the function of the HMUX sublayer to provide a multiplexing and demultiplexing function for channel aggregates of the slots within the TDM frame. The several services carried within each of these individual bearer channels have a distinct Ph_SAP that abstractly associates the service with a particular HMUX channel.

The PMD sublayer provides and delivers a bit stream; the PS sublayer maps this bit stream into a set of slots within a cyclic TDM frame, thereby providing the isochronous 8 kHz fundamental service; and the HMUX maps specific TDM frame slots into a set of channels each associated with a specific Ph_SAP. Thus, at the top of the HMUX sublayer, the service provided at each Ph_SAP by the PHY Layer is the acceptance and delivery of an isochronous bearer. This layered model of the internal operations of the PHY is graphically illustrated in figure 7-1. The functions of the HMUX, the PS, and PMD sublayers of the PHY are transparent to the Physical Layer service user; therefore, the services provided by these three sublayers will be characterized simply as the PHY service.

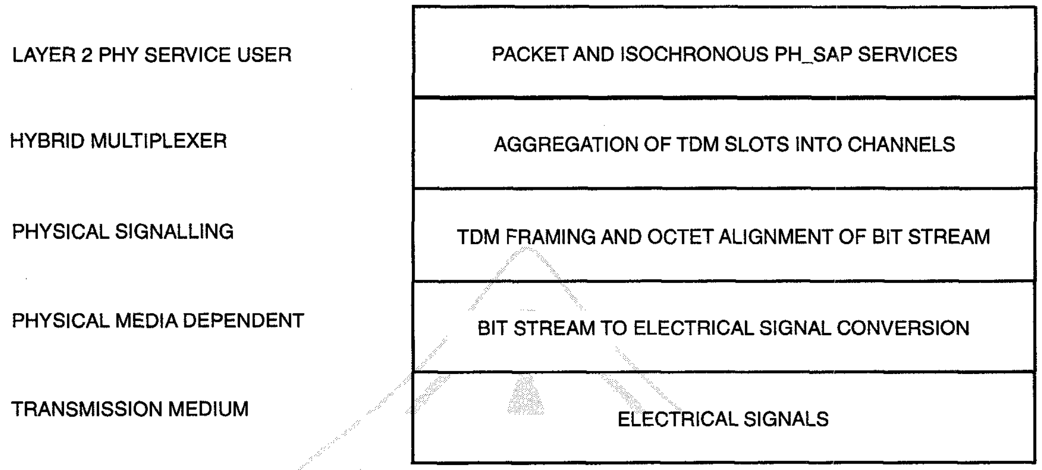


Figure 7-1—Sublayer model of the PHY service

The services provided by the PHY allow the PHY service user to send and receive information units, serialized as symbol streams. In the case of the MAC sublayer entity, the information units are MPDUs exchanged with a peer MAC sublayer entity. In the case of the LAPD entity, the information units are DL PDUs, defined in accordance with the DL framing rules of CCITT Q.921. In the case of sources of information sampled for transmission in an isochronous channel, the information units are octet sequences, which may be digitally encoded voice, video, and/or other information types.

As noted above, a timing indication shall be delivered to all users of the PHY service, but it is especially crucial for sources/sinks of information sampled for transmission in an isochronous channel.

7.1.5.3 PHY service access point identifiers

The PHY services are provided at the PHY service access points (PhSAPs) of the PHY service boundary. PhSAPs differentiate the access by the user to various services. PhSAP identification is detailed in this subclause.

PhSAPs are the architectural mechanism by which symbol streams are passed to the PHY service provider by the PHY service user, and to the PHY service user by the PHY service provider. The distinction of different PhSAPs at a single ISLAN PHY service boundary, either in an ISTE or in an AU, is required because that service boundary is used simultaneously to provide both packet-mode and circuit-mode PHY services. All PHY service users in that ISTE or AU are served at the same PHY service boundary.

It shall be a function of the layer management of the HMUX sublayer to provide each PHY service user with both the information and a Channel-Identifier that is mapped onto the PhSAP relevant to the service provided to that user. The PHY service user shall not attend to any other PhSAPs than those identified by the HMUX LME as relevant to its use of the PHY service.

Figure 7-2 presents a simple overview model of the way in which different PHY service users in the protocol stack of an ISTE or the AU attend to various subsets of the PhSAPs at the PHY service boundary in order to access different transmission services.

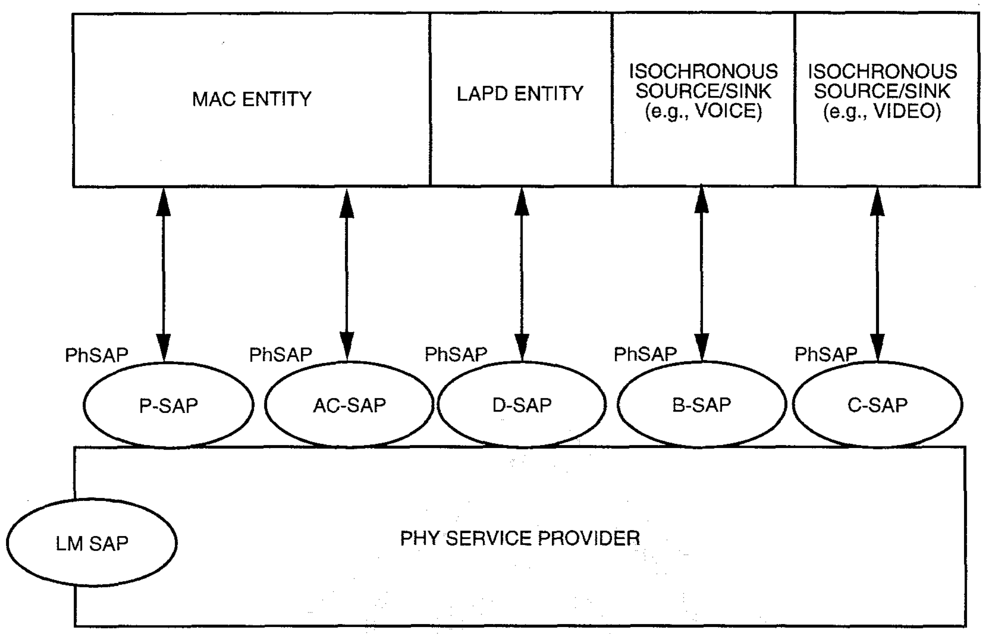


Figure 7-2—Overview model of PhSAP usage

PhSAPs shall be distinguished by binary identifiers which shall be locally administered across a given ISLAN interface at the PHY service boundary in an ISTE or in the AU. This administration ensures that PhSAPs function as abstract mechanisms, at a given PHY service boundary, through which transmission services are provided only to appropriate PHY service users, as determined by layer management. Local administration of PhSAP identifiers shall ensure that no single identifier is assigned to more than one PhSAP.

Aside from the needs of the local administration of PhSAP identification, a standard identification scheme for some of the PhSAPs is required for these purposes:

a) Identification of the PhSAPs to accomplish OSI layer and system management functions.

b) Definition of sets of PhSAPs to enable the architectural specification of various standard sets of services. These service sets will be provided at PHY service boundaries of different ISLAN interfaces; they will be provided to different PHY service users by different PHY service providers. For instance, the set of P-SAP and AC-SAP is the service set for the ISLAN MAC entity at any PHY service boundary.

The PhSAP is identified by a single value represented as a bit string, of length 16 (i.e., two octets); the assignment of all PhSAP-Id values is governed by rules set out in Chapter 9. For purposes of clarification, the identifier may be defined with the following structure:

PhSAP-Id::= BIT STRING (SIZE(16))

There is one series of PhSAPs within the 16-bit PhSAP space. The first four identifiers have the special function of providing for interactions between the PHY and the PHY layer management. All PhSAP identifiers provide for data users of the PHY service (conceptually those protocol layer entities above the PHY in the protocol stack, e.g., the MAC entity or an isochronous source/sink).

The following standard PhSAP identifications are expressed as specific bit string values (represented in hexadecimal notation). These strings shall be used at each PHY service boundary for the purpose of symbol exchange between the PHY service data users, PHY layer management, and the PHY service provider:

SYN-SAP	Ph-SAP-Id::='0000'H
TDM-MTN-SAP	Ph-SAP-Id::='0001'H
HMC-SAP	Ph-SAP-Id::='0002'H
RES-SAP	Ph-SAP-Id::='0003'H
D-SAP	Ph-SAP-Id::='0004'H
B1-SAP	Ph-SAP-Id::='0005'H
B2-SAP	Ph-SAP-Id::='0006'H
AC-SAP	Ph-SAP-Id::='0007'H
P-SAP	Ph-SAP-Id::='0008'H

The PhSAP value subrange, '0009'H through 'FFFF'H, is available for dynamic layer management assignment to identify various PhSAPs at each PHY service boundary through the exercise of OSI system and layer management functions.

Each of these PhSAPs identify a specific 8 kHz bearer of an octet that is encoded as defined in 6.5. The cyclic transmission and delivery of the SYN-SAP is the means by which the isochronous PHY service's 8 kHz fundamental timing signal is conveyed. The use of the individual bits and fields of bits within each of these PhSAPs is summarized below:

SYN-SAP Ph-SAP-Id::= '0000'H

Bits 0-6 Frame alignment word (FAW) generated within the PS sublayer as specified by the PHY layer management signal, ISTE-NOT-AU. The received FAWs are passed to the data user of the PHY service.

Bit 7 Reserved for a TDM multiframing function. This bit shall be accepted from the sending PHY service data user and delivered as received to the PHY data user. The management and control exerted on this bit by the sending and receiving PS sublayer is a matter for further study.

TDM-MTN-SAP Ph-SAP- Id::= '0001'H

Bits 0-7 All bits in the octet conveyed with this PhSAP identifier are a function of the layer management actions applied to the operations of the PS sublayer. Data user request information with this PhSAP identifier is ignored by the PHY service. Data user indications with this PhSAP identifier are a copy of the information received by the PS sublayer.

HMC-SAP Ph- SAP-Id::= '0002'H

Bits 0-4 These bits and subfields are a function of the layer management actions applied to the operations of the HMUX sublayer. Data user request information within these bits and subfields having this PhSAP identifier is ignored by the PHY service. Data user indications within these bits and subfields having this PhSAP identifier are a copy of the information received by the PS sublayer from its peer entity.

Bits 5-6 Reserved for future standardization use by IEEE 802.9. These bits shall be accepted from the sending PHY service data user and delivered as received to the PHY data user subject to the HMUX sublayer management operation that may be defined.

Bit 7 Data user request information in this bit having this PhSAP identifier is ignored by the PHY service. Data user indications with this PhSAP identifier are a copy of the information received by the HMUX sublayer from its peer entity. This bit reflects the actions of the sending HMUX LME.

RES-SAP Ph-SAP-Id::= '0003'H

Bits 0-7 Reserved for future standardization use by IEEE 802.9. This octet shall be accepted by a request from the sending PHY service data user and delivered as an indication as received to the PHY data user subject to further study and definition of the operation of the PHY.

D-SAP Ph-SAP-Id::= '0004'H

Bits 0-7 The function of this PHY service data user PhSAP is to provide an ISDN compatible D channel signalling bearer service. Its operation as a 2-bit or 8-bit service channel is dependent upon layer management actions conveyed in bit 7 of the HMC-SAP (Ph-SAP-Id::= '0002'H).

B1-SAP Ph-SAP-Id::= '0005'H

B2-SAP Ph-SAP-Id::= '0006'H

Bits 0-7 These two PHY service data user PhSAPs are unaffected by any PHY Layer functionality and are solely for the purpose of providing two ISDN-compatible 64 kb/s bearer channels.

AC-SAP Ph-SAP-Id::= '0007'H

Bits 0-7 The function of this PHY service data user PhSAP is to provide a bearer for the IEEE 802.9 ISLAN access control service. The details of the bits and subfields contained within the octets carried over this PhSAP are described in the MAC service specification.

P-SAP Ph-SAP-Id::= '0008'H

Bits 0-7 The function of this PHY service data user PhSAP is to provide a bearer channel for the conveyance of IEEE 802.9 MAC service data units. The details of information carried over this PhSAP are described in the MAC service specification.

In the IEEE 802.9 ISLAN, most signals carried in layer management (LM) channels are not transferred "peer-to-peer." Instead they are of local significance. In those cases where there is a LM-to-LM transfer of information [e.g., HMUX (DCR), TDM (RLB)] the information is subject to a degree of processing to improve the integrity of received signals and it is therefore inappropriate to treat the signals carried within these LM PhSAPs as "normal" primitives. The slots that convey LM signals are properly consistent with an OSI "request" and "indication" model. Therefore, it is at this level that the transfer of LM information is described as conformant with the OSI model of a PhSAP.

Table 7-1 relates PhSAPs with the TDM slot numbers and identifies the routing of the information within each service to the service boundary (SB) or LM, or both.

Table 7-1—Relationship of PhSAPs with the TDM slot numbers

TDM slot	Ph-SAP number	Passed to	Purpose and description
0	0	LM and SB	LM control of multiframing
1	1	LM and SB	LM for TDM maintenance
2	2	LM and SB	LM of HMUX and DCR control
3	3	LM and SB	LM (hysteresis) "Reserved"
4	4	SB	D channel
5	5	SB	B1
6	6	SB	B2
7	7	SB	Access control (AC) channel
8	8	SB	P channel
9 to 04FF	9 to FFFF	SB	"C" CCITT Q.931 assigned via HMUX

7.1.5.4 Detailed specification of the PHY service primitives

The service primitives used in the PHY service for all of the sources/sinks of information units are as follows:

— Ph-DATA request
— Ph-DATA indication

The Ph-DATA request and Ph-DATA indication primitives are used for data transfer by and to all PHY service users, as explained in detail below.

7.1.5.4.1 Ph-DATA request

This primitive provides the service of a PHY service user's transferring one symbol unit (i.e., a string of one or more symbols) to the PHY service provider at a specific PhSAP, in order to accomplish the user's transmission of a stream of symbols to its transmission peer. This stream is the serialization of an information unit.

Semantics of the service primitive

```
Ph-DATA request::= SEQUENCE
                        {
                         phsap [0] PhSAP-Id,
                         sdu     [1] Service-Data-Unit,
                         ft       [3] sdh-time-signal
                        }

PhSAP-Id::= EXTERNAL     -- See 7.1.1.5.3.
Service-Data-Unit::= SEQUENCE SIZE(8) OF Symbol
Symbol::= CHOICE   {
                   [0] Binary-0,
```

```
                        [1] Binary-1
                       }
Binary-0::= Symbol-Type
Binary-1::= Symbol-Type
Symbol-Type::= sdh-time-signal
```

The phsap parameter is the identifier for the local PhSAP, across which the primitive is passed.

The sdu parameter consists of eight symbols forming a single octet-wide slot within the cycle. It provides an elemental unit of 64 kb/s of channel bandwidth. The number of times that the primitive is sent during a cycle is either fixed in this standard, is implementation dependent, or is a function of the allocation of channels according to the procedures specified in 9.2.6 and Chapter 11.

The sdh-time-signal is an optional parameter which shall only be used at a PHY service boundary in the AU to input a timing signal from a backbone ISDN representing the clock source of the Synchronous Digital Hierarchy of a public network. The data value of this parameter is implementation dependent, since its timing is all that is relevant to the semantics of the parameter.

The symbol choice element has a binary value. The symbol value subset for all users of the Ph-DATA request service (both packet data sources/sinks and sources/sinks of information sampled for isochronous transmission) is {Binary-0, Binary-1}.

When generated

The PHY service user shall send the PHY service provider a Ph-DATA request primitive for the purpose of transferring a sequence of symbols to the peer (in either the ISTE or the AU) of the PHY service user, generating a Ph-DATA indication therein.

A Ph-DATA request primitive shall be sent by the PHY service user to the PHY service provider a fixed number of times during each successive TDM cycle (of length 125 μs). The number of times that the primitive is sent during a cycle shall be a function of the transmission speed of the channel to the PhSAP through which the PHY service is provided.

In the ISTE, a PHY layer management TDM frame loopback option may be used to permit independent contradirectional operation of the peer send and receive paths, or by default cause a two-octet delay between the receipt of a Ph-DATA indication and the sending of a Ph-DATA request by the PHY service user.

Effect of receipt

The PHY entity (more specifically, the HMUX sublayer entity) shall multiplex each symbol received in a Ph-DATA request into the next available symbol slot in the transmit stream of the channel which corresponds to the PhSAP to which the primitive has been submitted.

Additional comments

The synchronization of the request service to the indication service is the mechanism by which the synchronicity of the PHY service is achieved.

The PHY service is necessarily synchronous, in the sense that the PHY service user must always submit a Ph-DATA request primitive containing a symbol sequence when it receives a Ph-DATA indication primitive containing a symbol sequence. Thus symbol passage from the user to the provider of the PHY service is driven by a clock signal implied in the arrival of each Ph-DATA indication primitive.

A packet data user of the PHY service must pass a symbol sequence signifying, from the user's perspective, an idle pattern in its Ph-DATA request primitive whenever it receives a Ph-DATA indication but it is not in the process of transmitting the symbol stream of a data PDU. Therefore, the synchronicity of the PHY service does not prevent its being used for asynchronous data transmission.

The use of the sdh-time-signal parameter at the PHY service boundary in the AU is completely independent from the foregoing account of the synchronization of the submission of a Ph-DATA request primitive to the reception of a Ph-DATA indication primitive.

The description of the synchronization of the Ph-DATA request service by the Ph-DATA indication service does not imply, and is not intended to suggest, that there is some sort of application connection between the symbols transferred in each direction at any given time. The Ph-DATA request and Ph-DATA indication symbol sequences flowing across one PHY service boundary need have no connection with one another so far as the definition of the PHY service is concerned.

A source/sink of information sampled for isochronous transmission is always conveyed in the IEEE 802.9 PHY service as a multiple of 64 kb/s. A source/sink of information may pass a symbol signifying idle in its Ph-DATA request or indication primitive; however, it is important to note that the encoding and interpretation of such symbols is outside the scope of the IEEE 802.9 PHY service. For example, this could occur if the source were transmitting symbols at a user information rate that was not equal to the full transmission rate [e.g., adaptive differential pulse code modulation (ADPCM) encoded voice, or rate-adapted 19200 b/s over a 64 kb/s unrestricted digital transmission bearer service].

7.1.5.4.2 Ph-DATA indication

This primitive provides the service of the transfer of a symbol sequence from the PHY service provider to the PHY service user at a specific PhSAP, in order to accomplish the PHY service user's reception of a stream of symbols from its transmission peer. This stream is the serialization of an information unit.

Semantics of the service primitive

```
Ph-DATA indication::= SEQUENCE{
                              phsap[0] PhSAP-Id,
                              sdu   [1] Service-Data-Unit,
                              sv    [2] Status-Vector    OPTIONAL,
                             }

PhSAP-Id::= EXTERNAL     -- See 7.1.1.5.3.
Service-Data-Unit::= SEQUENCE SIZE(8) OF Symbol

Symbol::= CHOICE   {
                    [0] Binary-0,
                    [1] Binary-1,
                   }
Binary-0::= Symbol-Type
Binary-1::= Symbol-Type
Symbol-Type::= EXTERNAL

Status-Vector::= SEQUENCE SIZE(8) OF Status-Value
Status-Value::= ENUMERATED {valid(1), invalid(0)}
```

The PhSAP parameter is the identifier for the local PhSAP, across which the primitive is passed. The sdu parameter consists of eight symbols forming a single octet-wide slot within the cycle. It provides an elemental unit of 64 kb/s of channel bandwidth. The number of times that the primitive is received during a cycle is either fixed in this document, is implementation dependent, or is a function of the allocation of channels according to the procedures specified in 9.2.6 and Chapter 11.

The symbol parameter has a value of Binary-0 or Binary-1. The value subset for all users of the PHY service (both packet data sources/sinks and sources/sinks of information sampled for isochronous transmission) is {Binary-0, Binary-1}.

The value of the Status-Vector element is a sequence of indications, each of which indicates whether the value passed in the corresponding symbol member of the Service-Data-Unit element has been determined to be invalid or not by the functions of the PHY Layer.

When generated

The Ph-DATA indication primitive shall be sent to the service user a fixed number of times during each successive TDM cycle (of length 125 µs). The number of times that the primitive is sent during a cycle shall be a function of the transmission speed of the channel corresponding to the PhSAP through which the PHY service is provided.

Effect of receipt

When the PHY service user receives this primitive, the service user shall append to an information unit stream any subsequence of symbols in the Service-Data-Unit parameter which contains neither invalid symbols nor symbols of an idle pattern. The receipt of an invalid symbol indication shall trigger a notification to the PHY service user's LME. The receipt of an idle symbol pattern shall cause no action by the PHY service user. The receipt of a frame-timing value shall be used by a sink of information sampled for transmission in an isochronous channel (e.g., a sink of digitally encoded video information) to interpret the incoming symbol stream as an isochronous channel.

Additional comments

In any given Ph-DATA indication primitive, the symbol sequence may include symbols of an idle pattern, as determined by the application of the PHY service.

The reception of a succession of Ph-DATA indication primitives effectively provides a clock source signal to all sinks that are PHY service users. This allows a sink of information, sampled for transmission in an isochronous channel, to interpret the incoming symbol stream as octet aligned and to coordinate its sampling activity.

The provision of isochronous PHY service user timing, transmit/receive signal coordination, and alignment of octets is further enhanced by reference to the Ph-DATA indications delivered at PhSAP = '0000'H. These provide the fundamental 8 kHz timing needed by all other PhSAPs to construct SDUs for transmission, and reconstruct the timing and clocks associated with received PHY SDUs.

7.2 Convergence of services

The basic reference model for open systems interconnection [ISO 7498-4:1989 and CCITT Recommendation X.200 (1988)] specifies layer services in the protocol stack of an open system on the assumption that the services of the N Layer are provided by one protocol entity, the N Layer entity, to one user of those services, the N+1 Layer entity. Convergence of OSI layer services is introduced to address the architectural requirements, within the single protocol stack of an open system, to enable more than one N+1 Layer user to access a given N Layer service and to enable a given N+1 Layer user to make a selection among several distinct N Layer services.

The service architectures of the ISLAN MAC and PHY services are unique, among IEEE 802 LAN specifications, in the introduction of the architectural concept of the convergence of MAC and PHY services. The ISLAN MAC provides connectionless service for the transfer of LPDUs as well as the PDUs of any DL protocol comparable to LLC. The PHY provides a transparent symbol transfer service between peer ISTEs, regardless of the type of information being transferred (e.g. digitally encoded data, voice, video, or image).

Within the IEEE 802.9 architecture, provision has been made for convergence of multiple user services at the three different levels: Data Link Layer convergence (DLC), packetized service convergence (PSC), and PHY convergence.

DLC methodologies are outside the scope of IEEE 802.9, but for completeness and comparative purposes in selecting an appropriate convergence mechanism for specific applications of IEEE 802.9, a description of DLC over an ISLAN is provided.

PSC is provided in the ISLAN in order to support at the lowest level of OSI Layer 2 facilities to converge multiple packetized bearer services. These may include: an IEEE 802.9 MAC service in support of IEEE 802 LLC, an LAPD service to provide ISDN conformant signalling and packet services, and an LAPB service as defined in CCITT Recommendation X.25 (1993). In addition, architectural provision is made for the conveyance of the fixed length frames ("cells") employed by B-ISD.

Underneath the IEEE 802.9 MAC, provision is additionally made for the support of isochronous (non-packetized) services. The packet services (optionally subject to DLC or PSC convergence protocols) are conveyed in only one of several isochronous bearers known as the P channel. Other channels are converged (or aggregated) by the PHY HMUX sublayer into a cyclic TDM frame processed in the PHY's PS sublayer. The multiple channels supported by the PHY are provided at the PHY isochronous service interface defined in 9.2.4.

Each of these alternative convergence methods are presented in this clause. First an architectural overview of convergence concepts is used to set in context the descriptions of DLC, PSC, and PHY convergence that follow.

7.2.1 The concept of convergence

The two general forms of layer or sublayer service convergence are as follows:

a) The many (users)-to-one (provider) convergence of a single N Layer or sublayer data transfer service provider for various N+1 Layer or sublayer users, each with a distinct protocol

b) The one (user)-to-many (providers) convergence of multiple N Layer or sublayer data transfer services, each of which has a distinct protocol and is available to each one of a set of N+1 Layer or sublayer users

Figure 7-3 illustrates the two examples of convergence.

7.2.1.1 Many-to-one service convergence

The many-to-one convergence of an N Layer or sublayer service is the simultaneous provision of a common N Layer or sublayer service to multiple N+1 Layer or sublayer users of that service in the same open system, where the users differ in the N+1 Layer (or N+1 sublayer) protocols that they follow.

The many-to-one layer or sublayer service convergence is based upon the character of the N Layer (or N sublayer) service as the provider of data transfer to which service N+1 Layer (or N+1 sublayer) protocols are transparent.

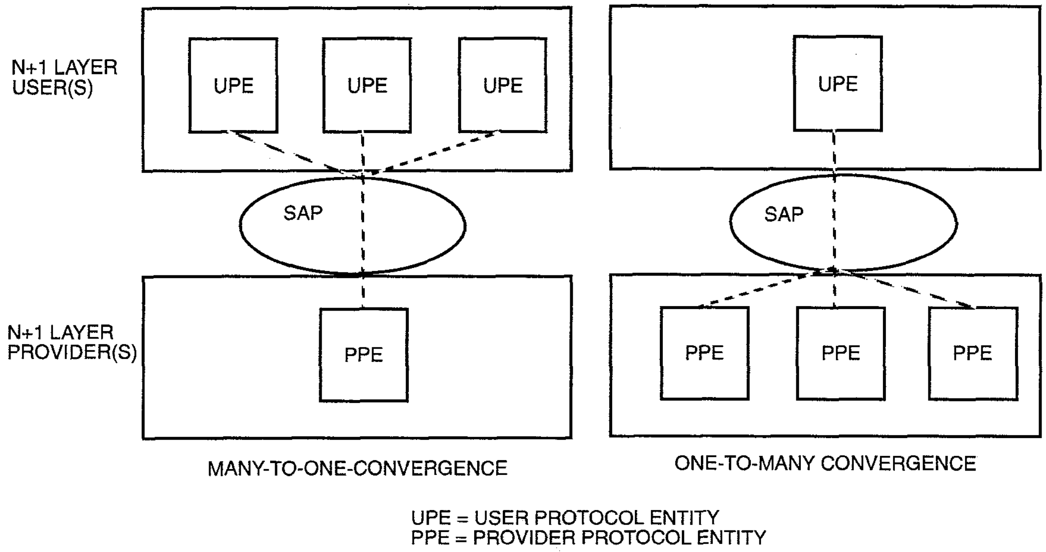


Figure 7-3—Two forms of service convergence

For instance, the LLC sublayer must be able to provide a convergence Logical Link service to an ISO/IEC 8208 Layer 3 user, a TCP/IP user, a CCITT Q.931 user, etc. The Logical Link service has been defined to be independent of the higher layer protocol elements encoded in the information field of the LPDU.

One mechanism for the provision of a many-to-one N Layer convergence service is the standard assignment of multiple N Layer SAPs, each corresponding to a different protocol suite of the user. For instance, standard LSAPs are assigned (see ISO/IEC TR 10178:1992) as follows:

a) z000 0000 Used in ISO/IEC 8802-2:1994 as the null address
b) z100 0000 Used by ISO/IEC DIS 15802-2 to indicate LLC sublayer management
c) z100 0010 Used in ISO/IEC 10038:1993 to identify the bridge spanning tree protocol
d) z101 0101 Used in ISO/IEC DIS 15802-2 to identify the SNAP SAP
e) z110 0000 Used in ARPANET to identify the internet protocol
f) z111 0000 Used in IEC 955:1989 to identify network management maintenance and initialization
g) z111 0001 Used in IEC 955:1989 to identify active station list maintenance
h) z111 0010 Used in ISO/IEC 9506: 1991 to identify manufacturing message service
i) z111 1110 Used to identify ISO/IEC 8208:1990 as the Network Layer protocol
j) z111 1111 Used to identify ISO/IEC TR 9577:1993

However, it is sometimes inappropriate to multiply SAPs for the purpose of many-to-one convergence; for instance, in the case of the MAC service, a unique MSAP address assignment in a given open system, corresponding to a single PHY entity (e.g., an adapter card), is a requirement for many applications. To avoid the need for multiple SAPs, IEEE 802.9 provides an additional mechanism for this architectural purpose: the identification of the protocol of the N+1 Layer user, of the N Layer convergence service, in each service primitive which passes across the SAP. In this MAC convergence example, service identification is provided by the SID parameter specified in the ISLAN MAC.

7.2.1.2 One-to-many service convergence

The one-to-many service convergence in which one N+1 Layer (or N+1 sublayer) user can access a number of distinct DL protocol service providers requires a way to distinguish various N SAPs, which correspond to different N Layer (or N sublayer) protocols.

One-to-many service convergence in the ISLAN is primarily a matter of interest at the DL service boundary. A number of Layer 3 protocol entities, within the OSI protocol stack of a given open system, can use each of a number of specific DL services offered at the DL service boundary of that same system. For instance,

a) CCITT Q.931 messages can be exchanged over LLC or LAPD connections.
b) ISO/IEC 8208 packets can be exchanged over LLC connection-mode or connectionless data transfer services or over LAPD connections.
c) A CCITT X.25 Layer 3 packet can be transmitted over either an LAPD or an LAPB connection.

Since, as described above, a number of standard assigned LSAPs for the LLC service are differentiated by the user protocol, it is necessary for one-to-many DL service convergence that there be a range of generic DLSAPs corresponding to the protocol of the service provider, i.e., the specific DL protocol. Thus, to follow this example, there must be one DLSAP in a one-to-one correspondence with the LLC protocol itself, which, in turn, defines a number of LSAPs that correspond to the protocol of the Layer 3 user of this specific DL service. For instance, a TCP/IP user of the generic DL service cannot select the LLC connectionless service by simply specifying the TCP/IP relevant LSAP (i.e., z110 0000) in passing the DL-UNITDATA request service primitive, specified in the LLC protocol (ISO/IEC 8802-2: 1994). This is because it cannot be guaranteed that the identifier of this LSAP is distinct in value from the values of the identifiers of other DL specific SAPs, corresponding to specific DL protocols other than LLC.

The generic DLSAPs just described must be entirely and logically separate from the range of LSAPs, as well as the ranges of SAPs of other specific DL protocols, e.g., the service access point identifiers (SAPIs) of LAPD.

7.2.2 DL service convergence in the ISLAN

7.2.2.1 Overview of DL service convergence

DL service convergence in the ISLAN is crucial because this convergence is a requirement for the protocol integration in an open system, in order to access both ISO/IEC 8802 LAN services and ISDN bearer services and teleservices.

The specification of DL service convergence shall employ the service definition of ISO/IEC 8886:1992 [which is identical, in all respects relevant to this discussion, to the corresponding sections of CCITT Recommendation X.212 (1988)]. ISO/IEC 8886:1992 is the only generic DL service specification available. Although ISO/IEC 8886:1992 is not a superset of other internationally standardized DL service and protocol specifications (in fact, it is not even a perfect abstraction of any one of them), its service in an ISLAN protocol stack is mapped into the services of LLC, LAPD, etc., by a DL service convergence function.

It is beyond the scope of this standard to define, in detail, how the mapping from the ISO/IEC 8886:1992 service to the services of specific DL protocols will be accomplished.

7.2.2.2 ISO/IEC 8886:1992 DL service primitives

Each Data-Link-Connection (DLC) service user may make a selection among a number of available, but protocol-distinct, DLC connection-mode services. This is done by specifying both the user's Layer 3 protocol and the provider's specific DL protocol, in conjunction with (but not included in) the following ISO/IEC 8886:1992 service primitives:

— DL-CONNECT (request, indication, response, and confirm)
— DL-DATA (request and indication)
— DL-RESET (request, indication, response, and confirm)
— DL-DISCONNECT (request and indication)

The same is true for the DL connectionless service, which uses the following ISO/IEC 8886:1992 service primitive:

— DL-UNITDATA (request and indication)

Figure 7-4 presents an SDL block model of the provision of the DL convergence service. The DLC_CONVERGENCE_FUNCTION block contains the function of mapping from the DL service primitives to the service primitives for each of the LLC, LAPD, and LAPB[13] service blocks, and vice versa.

7.2.2.3 Detailed specification of the DL service convergence structures

In the material that follows, the information structures are defined, using ASN.1 productions (not taken from ISO/IEC 8886:1992), which make possible both many-to-one and one-to-many DL service convergence.

The SDL signals, indicated in figure 7-4 as passed in the DLC_SB_1 and DLC_SB_2 channels between DL service users and DL service providers, are defined here as ASN.1 values.

```
dlc_p DL-SB-Signal::= {-- incomplete --}
dlc_u DL-SB-Signal::= {-- incomplete --}
```

The DL-SB-Signal contains both a DL service provider protocol specifier and a DL service user protocol specifier to make possible both many-to-one and one-to-many DL service convergence.

```
DL-SB-Signal::=
SEQUENCE      {
               [0]  CHOICE        {
                                   [0]  DL-Provider-Service-Primitive,
                                   [1]  DL-User-Service-Primitive
                                  },
               [1]  DL-Protocol-Specifier,
               [2]  DL-User-Protocol-Specifier
              }

DL-Protocol-Specifier::=
  ENUMERATED {llc(0),lapd(1),lapb(2)}
DL-User-Protocol-Specifier::=
  ENUMERATED {is8208(0),x25(1),q931(3)}
```

The DL-Provider-Service-Primitive differs from the DL-User-Service-Primitive in regard to the provider's service role. The role of the DL-Provider-Service-Primitive is to deliver an OSI service primitive indication or an OSI service primitive confirm to the DL service user. The role of the DL-User-Service-Primitive is to make an OSI service primitive request or an OSI service primitive response to the DL service provider.

```
DL-Provider-Service-Primitive::=

SEQUENCE          {
                   [0]  DL-Service-Primitive-Descriptor,
                   [1]  Service-Primitive-Role (indication | confirm)
                   [2]  DL-Service-Parameters
                  }

DL-User-Service-Primitive::=
```

[13]Throughout this discussion of the architecture of OSI service convergence in the ISLAN, three DL protocols are referenced: LLC, as defined in ISO/IEC 8802-2:1994, LAPD, as defined in CCITT Recommendations Q.920 (1993) and Q921 (1993); and LAPB, as defined in CCITT Recommendation X.25 (1993). However, only LLC and LAPD are supported in the ISTE protocol stack. LAPB is only one of a number of specific DL protocols which may optionally be implemented in this stack. Other optional DL protocols would include HDLC, SDLC, or LAPD+.

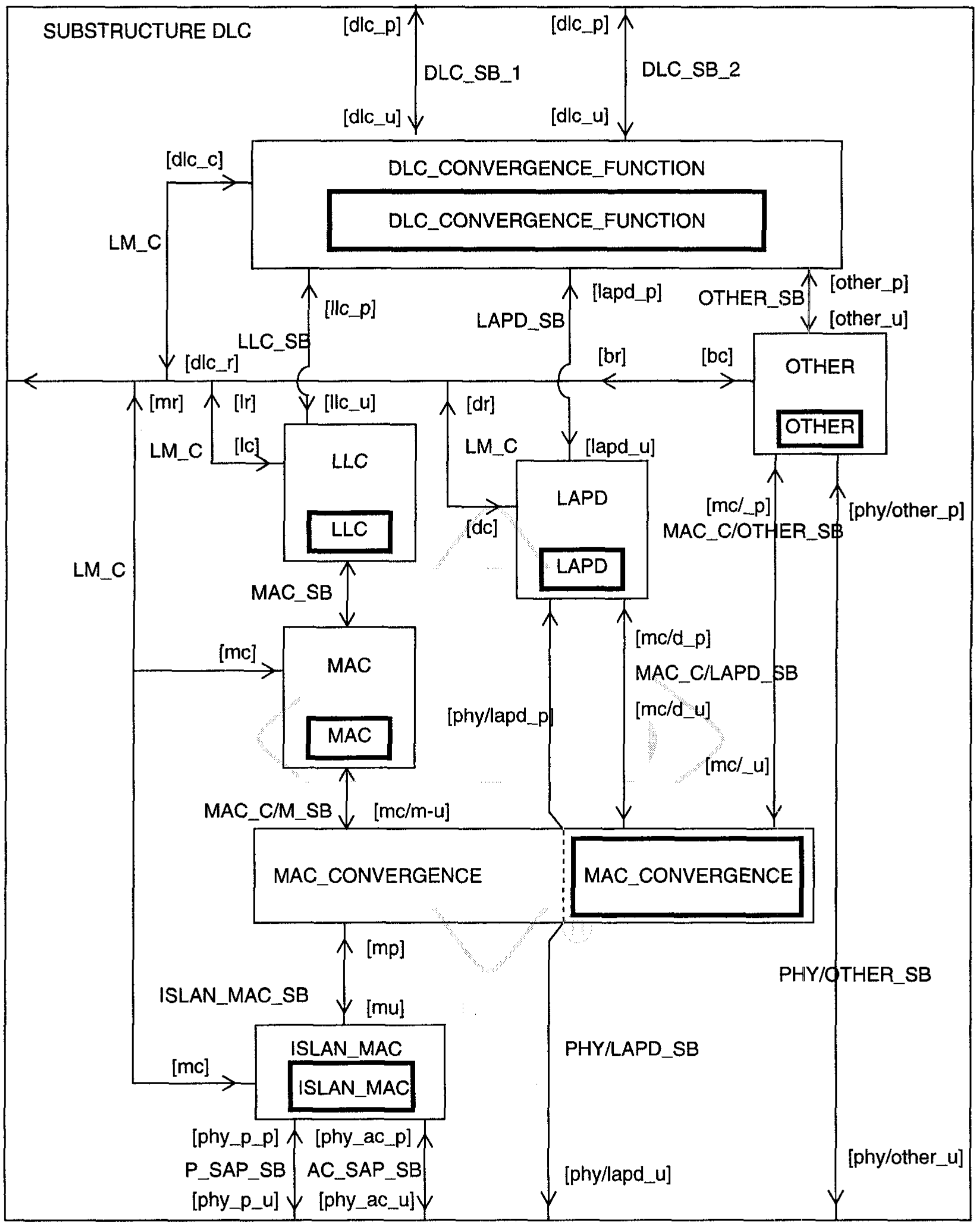


Figure 7-4—SDL block diagram—DL convergence service provision

```
SEQUENCE          {
                   [0]   DL-Service-Primitive-Descriptor,
                   [1]   Service-Primitive-Role (request | response)
                   [2]   DL-Service-Parameters
                  }
```

The DL service primitives fall either into the category of connection-mode primitives or the category of the connectionless (cnls) service primitive.

```
DL-Service-Primitive-Descriptor::=

CHOICE        {
               dl-cons-service-primitive [0]
                SEQUENCE                {
                                         CEI OPTIONAL,
                                         DL-Service-Primitive-Type
                                          (dl-connect | dl-data |
                                          dl-reset | dl-disconnect)
                                        }
               dl-cnls-service-primitive [1]
                DL-Service-Primitive-Type (dl-unitdata)
              }

DL-Service-Primitive-Type::= ENUMERATED
                                        {
                                         dl- connect(0),
                                         dl-data(1),
                                         dl-reset(2),
                                         dl-disconnect(3),
                                         dl-unitdata(4)
                                        }

Service-Primitive-Role::= ENUMERATED:{
                                         request(0), indication(1),
                                         response(2), confirm(3)
                                        }
```

The Connection-Endpoint-Identifier (CEI) is a means of distinguishing a number of connections through one DLSAP being used to access connection-mode service.

```
CEI::= OCTET STRING (SIZE(1))
```

The following ASN.1 production defines the parameter sets for the various DL service primitives, further qualified by service roles.

```
DL-Service-Parameters::=

CHOICE        {
               [dl-connect-ind] SEQUENCE
                                           {
                                            [0]   Called-Address,
                                            [1]   Calling-Address,
                                            [2]   QOS-Parameter-Set
                                           },

               [dl-connect-con] SEQUENCE
                                           {
                                            [0]   Responding-Address,
                                            [1]   QOS-Parameter-Set
                                           },

               [dl-data-ind] User-Data,
               [dl-reset-ind] SEQUENCE
                                           {
                                            [0]   Originator,
                                            [1]   Reset-Reason
                                           }

               [dl-reset-con] NULL,
               [dl-disconnect-ind] Originator,
               [dl-unitdata-ind] SEQUENCE
                                           {
```

```
                                    [0]  Source-Address,
                                    [1]  Destination-Address,
                                    [2]  QOS-Parameter-Set,
                                    [3]  User-Data
                                   },
        [dl-connect-req] SEQUENCE
                                   {
                                    [0]  Called-Address,
                                    [1]  Calling-Address,
                                    [2]  QOS-Parameter-Set
                                   },

        [dl-connect-res] SEQUENCE
                                   {
                                    [0]  Responding-Address,
                                    [1]  QOS-Parameter-Set
                                   },

        [dl-data-req] User-Data,
        [dl-reset-req] Reset-Reason,
        [dl-reset-res] NULL,
        [dl-disconnect-req] Disconnect-Reason,
        [dl-unitdata-req] SEQUENCE
                                   {
                                    [0]  Source-Address,
                                    [1]  Destination-Address,
                                    [2]  QOS-Parameter-Set,
                                    [3]  User-Data
                                   }
      }
```

The following ASN.1 value definitions specify the tag number values used by DL-Service-Parameters, above.

```
dl-connect-ind INTEGER::=       0
dl-connect-con INTEGER::=       1
dl-data-ind INTEGER::=          2
dl-reset-ind INTEGER::=         3
dl-reset-con INTEGER::=         4
dl-disconnect-ind INTEGER::=    5
dl-unitdata-ind INTEGER::=      6
dl-connect-req INTEGER::=       7
dl-connect-res INTEGER::=       8
dl-data-req INTEGER::=          9
dl-reset-req INTEGER::=        10
dl-reset-res INTEGER::=        11
dl-disconnect-req INTEGER::=   12
dl-unitdata-req INTEGER::=     13
```

The DLSAP identifiers or addresses are unspecified octet strings. Section 6 of ISO/IEC 8886:1992 states that DLSAP addresses are not unique in a global address space. They have local significance within a specific network. The DLSAP-ID could, however, be the one-octet LSAP or the two-octet SAPI.

```
Called-Address::=          DLSAP-Id
Calling-Address::=         DLSAP-Id
Responding-Address::=      DLSAP-Id
Source-Address::=          DLSAP-Id
Destination-Address::=     DLSAP-Id
DLSAP-Id::=                OCTET STRING
```

User-Data is the user information component of the DL frame. This field is of variable and unspecified length; the protocol payload in this field is transparent to the DL service and protocol.

```
User-Data::= OCTET STRING
```

The Originator parameter indicates the source of a DL-DISCONNECT indication or a DL-RESET indication.

```
Originator::= ENUMERATED
      {dls-user(0),dls-provider(1),unknown(2)}
```

The remaining parameters of the DL service primitives need not be defined further here, since the specification of a full DL service is beyond the scope of this discussion of convergence and the scope of this standard.

```
QOS-Parameter-Set::= EXTERNAL
Reset-Reason::= EXTERNAL
Disconnect-Reason::= EXTERNAL
```

The DL service primitives referenced here are not those of any particular DL protocol (e.g., ISO/IEC 8802-2:1994 LLC); rather they are the generic primitives specified in ISO/IEC 8886:1992.

7.2.2.4 SDL flow in the DLC substructure

In figure 7-4, the SDL channels (LLC_SB, LAPD_SB, and other_SB) represent the service boundaries between the specific DL protocol entities [represented by the SDL blocks (LLC, LAPD, and other)] and any Layer 3 users of that service. As such, these channels carry signals containing the service primitives defined in each of the respective DL protocol suites, with all associated parameters. The architecture of service convergence demands that each of these service boundaries of the specific DL protocol entities have the SDL block, DLC_CONVERGENCE_FUNCTION, in the N+1 Layer position in the protocol stack. Of course, DLC_CONVERGENCE_FUNCTION is in the N Layer position with respect to Layer 3 entities. As indicated above, DLC_CONVERGENCE_FUNCTION carries out the mapping between the generic DL service primitives exchanged with one or more Layer 3 entities and the service primitives exchanged with specific DL protocol blocks.

Each of the DL protocol blocks, in turn, exchanges service primitives using signals carried within the channels represented, in figure 7-4, as running vertically below the specific DL blocks: MAC_SB, MC_SB, PHY/LAPD_SB, MAC_C/LAPD_SB, MAC_C/OTHER_SB, and PHY/OTHER_SB.

7.2.3 MAC service convergence

7.2.3.1 Overview of MAC service convergence

At this point, the ISLAN architecture introduces what, with respect to Layer 3 PDU data transfer, may be characterized as an alternative to DL service convergence.

In figure 7-4, a MAC_CONVERGENCE functional block is situated in the N Layer position with respect to the specific DL protocol blocks. MAC_CONVERGENCE represents the functions of a new level of service convergence, i.e., that of the connectionless data transfer service of the ISLAN MAC protocol entity. MAC service convergence is clearly an example of many-to-one convergence. Since the protocol stack of an ISTE or an AU shall contain only one MAC protocol entity (i.e., that governing the request/grant access control protocol, explained in Chapter 5), MAC service convergence cannot have a one-to-many function.

MAC service convergence is required for the LLC/MAC protocol entity, because the IEEE 802.9 architecture has been set up for MAC service provision and other additional Layer 2 protocols. Mapping of service primitives is required in this case to distinguish IEEE 802 MAC, LAPD, and other protocol frames.

The device that permits this capability is the service identifier; this is the first field of the payload that carries the ISLAN MAC frame.

MAC service convergence may be optional for some DL protocol entities (e.g., LAPD or LAPB). The option is represented in figure 7-4 by the fact that the SDL blocks corresponding to the LAPD and the other protocol entities have two channels running vertically down from them. For instance, LAPD is connected to both the PHY/LAPD_SB and the MAC_C/LAPD_SB channels. More abstractly, this means that LAPD (and similarly LAPB), as represented in figure 7-4, has two service boundaries at which it can access data transfer service: the PHY service boundary which provides a symbol transfer service and the MAC_CONVERGENCE functional block which offers encapsulation in octet-aligned MAC frames which are transferred in a connectionless service as defined above.

For the LAPD or the other entities, the decision between the alternatives offered by the PHY service and the MAC convergence service is made by layer management intervention.

7.2.3.2 Substructure of the ISLAN MAC service provision

In this standard, only the IEEE 802 MAC service provision is fully specified. Support for LAPD (and other services) in P_d (or other P_x) channels is a matter for further study. In figure 7-5 the relationship of the LLC, MAC, MAC_SERVICE_CONVERGENCE, and ISLAN_MAC blocks is shown.

The LLC/MAC SDL block provides ISO/IEC 10039:1991 service in support of the LLC entity. The MAC_SERVICE_CONVERGENCE block provides a service multiplexing function to the LLC/MAC, the LAPD, and other service blocks. The ISLAN_MAC provides ISLAN_MAC SDU frame delimiting and service access control functions.

Conceptually, in send operations, the MAC SDL block maps the signals from the LLC block (ISO/IEC 10039:1991 service parameters) into an LLC/MAC SDU, as defined in 6.6, and queues this SDU to be sent via the MAC_SERVICE_CONVERGENCE block to the ISLAN_MAC block. In receive operations, the LLC/MAC block accepts SDUs routed to it by the MAC_SERV_CONVERGENCE block and transforms these into ISO/IEC 10039:1991 conformant LPDUs. Note that the integrity of LLC/MAC SDUs is maintained in the LLC/MAC block by means of IEEE 802 specific CRC generators and checkers.

The MAC_SERVICE_CONVERGENCE block accepts requests, over the LLC/MAC_SID_SB channel, for transmission of LLC/MAC SDUs. Conceptually, these are queued along with any concurrent send requests for LAPD or Other SDUs. The MAC_SERVICE_CONVERGENCE block prioritizes such concurrent service requests, and delivers a sequential stream of ISLAN_MAC SDUs via the ISLAN_MAC_SB channel to the ISLAN_MAC block. The MAC_SERVICE_CONVERGENCE block passes ISLAN_MAC SDUs to the ISLAN_MAC block with a prepended service identifier (SID) extended information element.

In receive operations, the MAC_SERVICE_CONVERGENCE block processes the SID information element with which ISLAN_MAC SDUs commence. It uses the SID to route (demultiplex) the sequentially delivered ISLAN_MAC SDUs to the appropriate LLC/MAC, LAPD, or other service functional blocks.

Architecturally, the transmit functionality of the ISLAN_MAC block requires it to delimit ISLAN_MAC SDUs (from P channel idles) by using AC channel signals. Delimitation of the start of ISLAN_MAC frames uses that start of frame signal in the AC channel service. The end of an ISLAN_MAC SDU is defined by the size parameter in the ISLAN_MAC_SB channel. AC_PHY_SB signals are also used to synchronize the transmission of ISLAN_MAC SDUs in accordance with the details of the request/grant protocol defined in Chapter 8.

The ISLAN_MAC receive functionality requires incoming ISLAN_MAC SDUs to be delimited from the P channel idle signal.

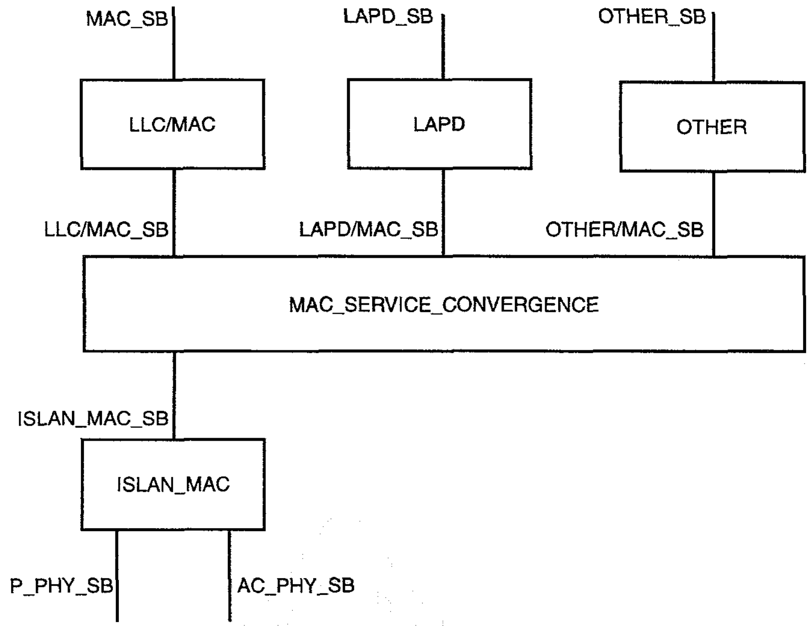


Figure 7-5—SDL block diagram—ISLAN MAC service provision

Since in this IEEE 802.9 architecture there is a complete specification of the LLC/MAC service only, the functions of the LLC/MAC, MAC_SERVICE_CONVERGENCE, and ISLAN_MAC blocks may be realized in a single block (or "compressed" stack). The model defined here provides the architectural facilities required to support the additional packetized services (e.g., LAPD, LAPB, "cell," or Other).

7.2.3.3 Detailed specification of the MAC service convergence structures

The signals carried in the MAC_SERVICE channel are defined here as ASN.1 values.

```
mac_p MAC-SB-Signal::= {-- incomplete --}
mac_u MAC-SB-Signal::= {-- incomplete --}
```

The MAC-SB-Signal contains a protocol specifier for the MAC service user, whether or not convergence is involved.

```
MAC-SB-Signal::=
SEQUENCE     {
             [0]   CHOICE       {
                                  [0]   DL-Provider-Service-Primitive,
                                  [1]   DL-User-Service-Primitive
                                },
             [1]   MAC-User-Protocol-Specifier
             }
MAC-User-Protocol-Specifier::=
 ENUMERATED {llc(0),lapd(1),lapb(2)}
```

The ISO/IEC 10039:1991 service definition contains no element corresponding to a MAC service user protocol specification (defined in MAC-User-Protocol-Specifier above). However, this does not imply that this standard is extending the MAC service specification. ISO/IEC 10039:1991 defines a core MAC service, which implies nothing about the existence of supplements to that service. In any case, it would be impossible to define MAC service convergence appropriately without the protocol specifier for the user of the service. Without this user protocol specification in the MAC-SB-Signal, the routing of the signals presented in figure 7-5 would be impossible. OSI layer management does not provide a function that could accomplish the same objective.

The MAC-Provider-Service-Primitive differs in data type definition from the MAC-User-Service-Primitive only in regard to the role of the service primitive. The provider MA-UNITDATA primitive passes indications only, and the user MA-UNITDATA primitive passes requests only.

```
MAC-Provider-Service-Primitive::=

SEQUENCE            {
                     [0]   Service-Primitive-Role (indication),
                     [1]   MAC-Service-Parameters
                    }

MAC-User-Service-Primitive::=

SEQUENCE            {
                     [0]   Service-Primitive-Role (request),
                     [1]   MAC-Service-Parameters
                    }

-- Service-Primitive-Role is defined above.
```

The elements of the MA-Service-Parameters are explained in 7.1.

```
MA-Service-Parameters::=

SEQUENCE            {
                     [0]   Destination-Address,
                     [1]   Source-Address,
                     [2]   MSDU,
                     [3]   Priority
                    }

MSAP-Id::= OCTET STRING (SIZE(6))
Destination-Address::= MSAP-Id
Source-Address::= MSAP-Id

MSDU::= OCTET STRING
Priority::= INTEGER
```

7.2.3.4 Rationale for MAC service convergence

Both DL service convergence and MAC service convergence are included in the ISLAN service architecture. However, they may be seen as alternatives to provide very similar Layer 3 PDU data transfer functions. For instance, the very same CCITT Q.931 signalling message could be transferred either in an LLC PDU, encapsulated within an MPDU (using DL service convergence), or within an LAPD PDU, encapsulated within an MPDU (using MAC service convergence).

MAC service convergence is still required if DL service convergence is implemented, and vice versa. The implementation of DL service convergence involves a different set of considerations (involving issues of protocol overhead, existing designs, complexity of implementation, etc.) than the implementation of MAC service convergence. In particular cases, either type of convergence may offer advantages over the other, in regard to cost, manageability, etc. This standard does not require the implementation of both of these types of service convergence, or even one of them. The intention in specifying both of them is to provide the most generally useful architecture.

In any case, either DL service convergence or MAC service convergence could be used for several key applications in customer premises equipment design. For instance, through either type of convergence, CCITT Q.931 signalling could be accomplished by using ISO/IEC 8802 LAN services, instead of or in addition to physically out-of-band D channel packet services. The use of the ISLAN's ISO/IEC 8802 LAN services for signalling, assuming that one or more distributed ISDN PBX modules had a direct connection to an ISLAN (through their function as AUs presumably), could be a lower cost alternative to D channel interfaces between an AU and such a PBX. Furthermore, certain key end-user applications in the ISLAN may only be successful if there is a close application association between signalling information elements and user data carried through a LAN. This association could be achieved through the use of one or both of these types of packet-mode service convergence.

7.2.4 PHY service convergence

The DL service convergence (both many-to-one and one-to-many) and the MAC service convergence (many-to-one only) are concerned exclusively with packet-mode data transfer. On the other hand, the PHY Layer provides a many-to-one service convergence that is available for both packet-mode and circuit-mode data transfer at the ISLAN interface.

As explained in 7.1, the PHY service is provided across a range of PhSAPs which make the many-to-one service convergence possible. Since there is only one PHY protocol available at the ISLAN interface, there can be no one-to-many PHY service convergence.

The IEEE 802.9 support of isochronous (nonpacketized) services, in which the P channel is a bearer for the IEEE 802.9 packet services, are converged (or aggregated) within the HMUX sublayer of the PHY into an isochronous TDM frame that is conveyed from the PHY's PS sublayer service to its PHY peer for demultiplexing by the receiving HMUX sublayer.

Other channels are converged (or aggregated) into a cyclic TDM frame and provided at the PHY isochronous service interface as defined in 9.2.4. Note that the PHY isochronous service interface provides "virtual" multiplexing in the sense that each service octet transferred over the interface has an associated channel identifier distinguishing the separate services supported. This (de)multiplexing functionality is not so much real as it serves as a logical abstraction of the multiplexing function, which must be explicitly processed by the service user of the PHY isochronous service interface.

7.3 Management services

7.3.1 LME/MAC service

The term "LM-MA" is used to signify "layer management-to-MAC layer interface."

This subclause describes the services that the MAC sublayer entity provides to the local MAC layer management entity (MLME) to accomplish layer management functions. Some of these services enable the MLME to control the MAC through monitoring and intervention in MAC functioning. In the conceptual service model, these layer management control functions are accomplished by the exchange of service primitive messages between the MLME and the MAC. This service model is not intended to imply any limitation on implementations.

MAC connectionless communication services are provided to an MLME so that it can exchange MAC layer management SDUs with peer MLME entities.

This service specification complies with ISO 7498-4:1989. The following service primitives have been defined:

— LM-MA-GET
— LM-MA-RETURN
— LM-MA-SET
— LM-MA-ACTION
— LM-MA-STATUS
— LM-MA-NOTIFICATION
— LM-MA-UNITDATA request
— LM-MA-UNITDATA indication

The LM-MA-GET, LM-MA-RETURN, LM-MA-SET, LM-MA-ACTION, LM-MA-STATUS, and LM-MA-NOTIFICATION service primitives correspond to information elements in the managed object (MO) class specifications for 8029-AU-MAC-Entity and ISTE-MAC-Entity in Chapter 10.

The LM-MA service primitives and the MO specifications define two separate service boundaries, as shown in figure 7-6. (Also see figure 10-3.) The service primitives define services provided to the MLME by the MAC across the LM-MA service boundary. The MO specifications define services provided to a local agent system management application process (SMAP) across the LMI service boundary.

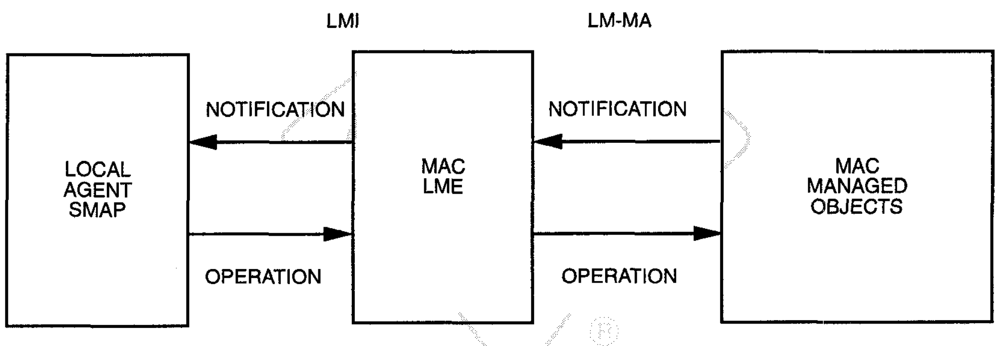


Figure 7-6—MLME service boundaries

The LMI services, provided by the MLME to the SMAP, are defined in Chapter 10 of this standard, in accordance with the OSI structure of management information (SMI) [ISO/IEC 10165-4:1992].

The correspondence between the services at these two boundaries is a reflection of the fact that the services provided to the MLME by the MAC, at the LM-MA service boundary, are used by the MLME to provide corresponding services to the SMAP at the LMI.

7.3.1.1 Detailed specifications of LME/MAC service primitives

7.3.1.1.1 LM-MA-GET

This service primitive provides an atomic read request service to the MLME.

Semantics of the service primitive

This service primitive is used to request the value of a single MAC attribute identified by a registered object identifier.

```
LM-MA-GET::= SEQUENCE          {
                                AttributeId
                               }

AttributeId::= MACAttributeObjId
MACAttributeObjId::= SEQUENCE OF INTEGER
```

When generated

This service primitive is passed from the MLME to the MAC when an SMAP requires the value of a particular MAC attribute for a management application. The SMAP uses the SMI (ISO/IEC 10165-4:1992) defined GET operation at the LMI to command the MLME to obtain an attribute value, and the MLME, in turn, passes the LM-MA-GET to the MAC.

Effect of receipt

When the MAC receives this service primitive, it returns the value of the attribute in question to the MLME as the parameter of an LM-MA-RETURN service primitive.

Additional comments

The LM-MA-GET is an atomic read request operation, meaning that it is performed on a single attribute. See Chapter 10 for the use of the GET operation in the specification of the attributes of a MAC MO.

7.3.1.1.2 LM-MA-RETURN

This service primitive returns a requested MAC attribute value to the MLME.

Semantics of the service primitive

This service primitive is used to return the value of an attribute identified, in a corresponding LM-MA-GET service primitive, by a registered object identifier.

```
LM-MA-RETURN::= SEQUENCE       {
                                AttributeValue
                               }
AttributeValue::= ANY
```

When generated

This service primitive is passed from the MAC to the MLME following the receipt by the MAC of an associated LM-MA-GET service primitive from the MLME.

Effect of receipt

When the MLME receives an attribute value in an LM-MA-RETURN service primitive, it then satisfies an outstanding GET service request issued by the SMAP at the LMI.

Additional comments

This service is conceived to be provided instantaneously to the MLME by the MAC, following the MAC's reception of an LM-MA-GET service primitive. The LM-MA-GET service primitives are conceived to be passed across the SAP sequentially. Thus there cannot be multiple outstanding LM-MA-GET requests at the same time, and so there is no need to include, as a parameter of the LM-MA-RETURN, an identifier of the attribute the value of which is returned.

7.3.1.1.3 LM-MA-SET

This service primitive provides an atomic write request service to the MLME.

Semantics of the service primitive

This service primitive is used to change the value of an attribute identified by a registered object identifier to a specified value.

```
LM-MA-SET::= SEQUENCE          {
                                 [0] AttributeId,
                                 [1] AttributeValue
                                }
AttributeId::= MACAttributeObjId
AttributeValue::= ANY
```

When generated

This service primitive is passed from the MLME to the MAC when an SMAP requires a change of value in an attribute of the MAC MO, in accordance with a management application. The SMAP uses the SMI (ISO/IEC 10165-4:1992) defined SET operation at the LMI to command the MLME to set an attribute to a specified value, and the MLME, in turn, passes the LM-MA-SET to the MAC.

Effect of receipt

When the MAC receives this service primitive, it changes the value of the attribute in question. Neither the attribute, identified in the service primitive, nor the ability to change it directly, belongs to the MLME.

Additional comments

The LM-MA-SET is an atomic write operation performed on a single attribute. See Chapter 10 for the use of the SET operation in the specification of the attributes of a MAC MO.

7.3.1.1.4 LM-MA-ACTION

This service primitive provides a service to the MLME by which it is able to initiate any of a defined set of actions in the already initialized MAC, at the direction of an SMAP.

Semantics of the service primitive

The parameter of this service primitive is an action descriptor chosen from a list of those actions that the MLME can initiate in the MAC to accomplish layer management tasks.

```
LM-MA-ACTION::= SEQUENCE            {
                                      Action
                                     }
Action::= CHOICE  {
                   activate    [0]   MACattributeSet,
                   deactivate  [1]   NULL,
                   ignoreGrant [2]BOOLEAN,
                   reset       [3]   MACAttributeSet,
                   -- other actions are for further study
                  }

MACAttributeSet::= SET OF MACAttributeAssertion
MACAttributeAssertion::= SEQUENCE    {
                                      [0]   MACAttributeObjId
```

```
                                        [1]  MACAttributeValue
                                      }
MACAttributeValue::= ANY
```

The parameter of this service primitive for the reset action is specified as an unordered list of MAC attribute value assertions, which define values for the attributes to be set, through the use of this service. Any attribute of the MAC, not included in this list, will be set to an already specified default value.

Each item in the unordered list, MACAttributeSet, is a pair. The first member of the pair is a registered attribute identifier, which is defined as a vector of integers. The second member of the pair is the value to which the attribute is to be set upon initialization.

When generated

This service primitive is passed from the MLME to the MAC when an SMAP requires an intervention in the operation of the MAC. The SMAP uses one of the SMI (ISO/IEC 10165-4:1992) defined ACTION operations at the LMI to command the MLME to initiate a specific action. The MLME, in turn, passes an LM-MA-ACTION service primitive to the MAC with a parameter corresponding to the ACTION operation requested at the LMI.

Effect of receipt

When the MAC receives the LM-MA-ACTION, it performs the action requested.

The deactivate action causes the affected MAC to suspend functioning so that other SMAP directed actions can be performed upon the MAC of its station's transmission peer (i.e., the AU MAC if the suspended MAC is in the ISTE, or vice versa). The deactivate action might be invoked so that diagnostic procedures could be executed in the MAC of its station's transmission peer.

The activate action causes a MAC to resume functioning if it has been in a suspended state due to having been subject to an earlier deactivate action. If the MAC that receives the LM-MA-ACTION (activate) is not in a suspended state, then the activate request has no effect upon it.

The ignoreGrant=YES action causes the MAC to ignore grant indications from the remote MAC of its station's transmission peer when the local MAC transmits MAC frames. The ignoreGrant=NO action has an opposite effect upon the affected MAC.

Upon receipt of the LM-MA-ACTION (reset) service primitive, the MAC resets all of its attributes, using the supplied parameter values and known defaults.

Additional comments

LM-MA-ACTION (reset) is used when the atomic attribute set request, provided above (LM-MA-SET), is not suitable because all or many of the MAC attributes must be set.

In general, actions invoked by LM-MA-ACTION request services are conceived of as affecting the whole operation of the MAC, rather than merely one attribute as is the case with the operations initiated by LM-MA-GET and LM-MA-SET.

7.3.1.1.5 LM-MA-STATUS

This primitive returns to the MLME the result status of an intervention in the MAC initiated by an LM-MA-ACTION being passed to the MAC.

Semantics of the service primitive

The parameters of this service primitive identify the action type and the result status returned by the MAC after an action has been performed.

```
LM-MA-STATUS::= SEQUENCE           {
                                    [0] ActionId,
                                    [1] Status
                                   }
ActionId::= ENUMERATED        {
                               act(0),
                               deact(1),
                               igGR(2),
                               res(3),
                               -- other actions are for further study.
                              }
Status::= ENUMERATED     {
                          success(0),
                          failedNoReason(1),
                          -- Other values are for further study.
                         }
```

When generated

The MAC passes this service primitive to the MLME after having received and satisfied the request of an LM-MA-ACTION from the MLME. In order to maintain conceptual simplicity in the service model, it is assumed that there is no overlapping of actions; no new LM-MA-ACTION is passed by the MLME to the MAC until any outstanding LM-MA-STATUS corresponding to an earlier LM-MA-ACTION has been returned to the MLME. This single instruction stream feature of the service model is not intended to imply any limitation on multithreaded implementations.

Effect of receipt

If the status value returned to the MLME is success, then no further action is defined for this service. If it is anything other than success, then the MAC shall repeat the request for service by passing the associated LM-MA-ACTION to the MAC again, with a maximum of two retries. If, after this number of retries, there is still no success status value returned, an appropriate failure notification is delivered by the MLME to the SMAP, across the LMI.

When the MLME receives an LM-MA-STATUS service primitive indicating success, it then completes the satisfaction of an outstanding ACTION service request issued by the SMAP at the LMI.

Additional comments

This service is conceived of as synchronous in the sense that, after passing an LM-MA-ACTION, the MLME remains in a wait state, so far as interactions with the MAC are concerned, until it receives an associated LM-MA-STATUS with the value success. However, the service specification does not depend, in any way, upon the realtime bounds (i.e., minimum and maximum timer values) of the wait state.

The synchronous feature of the service model is not intended to limit multithreaded implementations.

7.3.1.1.6 LM-MA-NOTIFICATION

This service primitive notifies the MLME of events of management significance which occur during MAC functioning.

Semantics of the service primitive

```
LM-MA-NOTIFICATION::= SEQUENCE      {
                                      Event
                                     }
Event::= ENUMERATED      {
                          rxResErrThr(0),
                          rxFcsErrThr(1),
                          invalidCommandReq(2),
                          failedAction(3)
                          -- Other values are for further study.
                         }
```

LM-MA-NOTIFICATION (RxResErrThr) is the notification that a threshold has been reached for the counter which holds the number of occasions in which an MPDU was received when the resources required for the reception were not available.

LM-MA-NOTIFICATION (RxFcsErrThr) is the notification that a threshold has been reached for the counter which holds the number of occasions in which an MPDU was received when the calculation of the frame check sequence indicated one or more errors in the bit sequence received.

LM-MA-NOTIFICATION (InvalidCommandReq) is the notification that the service user has requested a service that cannot be provided at this point (in processing) by the service provider.

LM-MA-NOTIFICATION (failedAction) is the notification that an LM-MA-ACTION has failed, as explained above.

When generated

This service primitive is passed to the MLME by the MAC when one of a defined set of events has occurred in the MAC.

Effect of receipt

When this service primitive is received by the MLME, it generates a corresponding notification to the SMAP. The syntax and the semantics of the specification of the notification to the SMAP at the LMI are defined in Chapter 10 of this standard, in accordance with the SMI (ISO/IEC 10165-4:1992).

Additional comments

This service is conceived of as asynchronous to the operation of the MLME (i.e., the MLME does not enter a prior wait state to receive the notification; rather it is interrupted for this purpose).

7.3.1.1.7 LM-MA-UNITDATA request

This service parameter allows one MLME to address and send MAC layer management SDUs (MLMSDUs) to the MLME of its station's transmission peer (AU or ISTE, as the case may be), using the connectionless MAC service for the exchange of MPDUs.

In all respects, this service functions in a way identical to the MA-UNITDATA request service specified in ISO/IEC 10039:1991.

Semantics of the service primitive

```
LM-MA-UNITDATA request::= SEQUENCE
                                          {
```

```
                                        [0] DestinationAddress,
                                        [1] SourceAddress,
                                        [2] MLMSDU,
                                        [3] Priority
                                       }
DestinationAddress::= MSAPAddress
SourceAddress::= MSAPAddress
MSAPAddress::= BIT STRING {Lsb(0),Msb(47)}
MLMSDU::= OCTET STRING
Priority::= INTEGER
```

The semantics of this service primitive are the same as those of the MA-UNITDATA request service primitive, as defined in ISO/IEC 10039:1991 and described above.

When generated

The local MLME passes this service primitive to its MAC when this MLME has a functional requirement to send an MLMSDU to the MLME of its station's transmission peer.

Effect of receipt

The MAC creates an MPDU containing the MLMSDU received from the MLME and attempts to transmit it as a bit stream at the MAC/PHY service boundary.

Additional comments

This service is only required for the synchronization of the operations of an MLME and its peer MLME in its station's transmission peer. It is not required directly for the management functions of the SMAP. An LM-MA-UNITDATA request is not passed to the MAC as a consequence of the MLMEs having been commanded to do so by their local agent SMAP.

In order for peer MLMEs to exchange MLMSDUs, there must be a dedicated, well understood group MAC address which the sender inserts into both the destination and source address fields of the LM-MA-UNITDATA request.

7.3.1.1.8 LM-MA-UNITDATA indication

This service parameter allows one MLME to be addressed and to receive MLMSDUs from the MLME of its station's transmission peer (i.e., the AU or an ISTE, as the case may be), using the connectionless MAC service for the exchange of MPDUs.

Semantics of the service primitive

```
LM-MA-UNITDATA indication::= SEQUENCE
                                       {
                                        [0] DestinationAddress,
                                        [1] SourceAddress,
                                        [2] MLMSDU,
                                        [3] Priority
                                       }
```

The semantics of this service primitive are the same as those of the MA-UNITDATA indication service primitive, as defined in ISO/IEC 10039:1991.

When generated

The local MAC passes this service primitive to its MLME when this MAC has received an MLMSDU for delivery to the MLME.

Effect of receipt

There is no further effect in the MLME, defined by this service, after receipt of the MLMSDU.

Additional comments

This service is only required for the synchronization of the operations of an MLME and its peer MLME in its station's transmission peer. It is not required for the management functions of the SMAP.

In order for peer MLMEs to exchange MLMSDUs, there must be a dedicated, well understood group MAC address which the receiver finds in both the destination and source address fields of the LM-MA-UNITDATA request.

7.3.2 LME/PHY service

The term "LM-PH" is used to signify the "layer management-to-Physical Layer interface."

This subclause describes the services which the Physical Layer entity (PHY) provides to the local PHY Layer management entity (PLME) to accomplish layer management functions. Some of these services enable the PLME to control the PHY through monitoring and intervention in PHY functioning. In the conceptual service model, these layer management control functions are accomplished by the exchange of primitive messages between the PLME and the PHY. This service model is not intended to imply any limitation on implementations.

This service specification complies with ISO 7498-4:1989.

The following service primitives have been defined:

— LM-PH-GET
— LM-PH-RETURN
— LM-PH-SET
— LM-PH-ACTION
— LM-PH-STATUS
— LM-PH-NOTIFICATION

The LM-PH-GET, LM-PH-RETURN, LM-PH-SET, LM-PH-ACTION, LM-PH-STATUS, and LM-PH-NOTIFICATION service primitives correspond to information elements in the MO class specifications for the IEEE 802.9 PHY-Entity in Chapter 10.

The LM-PH service primitives and the MO specifications define two separate service boundaries, as shown in figure 7-7. (Also see figure 10-3.) The service primitives define services provided to the PLME by the PHY across the LM-PH service boundary. The MO specifications define services provided to a local agent SMAP across the LMI service boundary.

The LMI services, provided by the PLME to the SMAP, are defined in Chapter 10 of this standard, in accordance with the SMI (ISO/IEC 10165-4:1992).

The correspondence between the services at these two boundaries is a reflection of the fact that the services provided to the PLME by the PHY, at the LM-PH service boundary, are used by the PLME to provide corresponding services to the SMAP at the LMI.

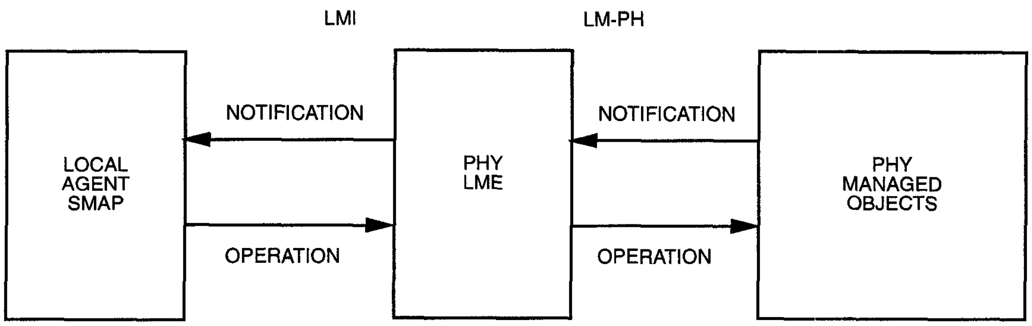


Figure 7-7—PLME service boundaries

7.3.2.1 Detailed specification of the LME/PHY service primitives

7.3.2.1.1 ML-PH-GET

This service primitive provides an atomic read request service to the PLME.

Semantics of the service primitive

This service primitive is used to request the value of a single attribute of the PHY MO (or of the HMUX, PS, or PMD MOs) identified by a registered object identifier.

```
LM-PH-GET::= SEQUENCE    {
                          AttributeId
                         }
AttributeId::= PHYAttributeObjId
PHYAttributeObjId::= SEQUENCE OF INTEGER
```

When generated

This service primitive is passed from the PLME to the PHY when an SMAP requires the value of a particular PHY attribute for a management application. The SMAP uses the SMI (ISO/IEC 10165-4:1992) defined GET operation at the LMI to command the PLME to obtain an attribute value, and the PLME, in turn, passes the LM-PH-GET to the PHY.

Effect of receipt

When the PHY receives this service primitive, it returns the value of the attribute in question to the PLME as the parameter of an LM-PH-RETURN service primitive.

Additional comments

The LM-PH-GET is an atomic read request operation, meaning that it is performed on a single attribute. See Chapter 7 for the use of the GET operation in the specification of the attributes of a PHY MO (or more specifically, of the HMUX, PS, or PMD MOs).

7.3.2.1.2 LM-PH-RETURN

This service primitive returns a requested PHY attribute value to the PLME.

Semantics of the service primitive

This service primitive is used to return the value of an attribute identified, in a corresponding LM-PH-GET service primitive, by a registered object identifier.

```
LM-PH-RETURN::= SEQUENCE        {
                                 AttributeValue
                                }
AttributeValue::= ANY
```

When generated

This service primitive is passed from the PHY to the PLME following the receipt by the PHY of an associated LM-PH-GET service primitive from the PLME.

Effect of receipt

When the PLME receives an attribute value in an LM-PH-RETURN service primitive, it then satisfies an outstanding GET service request issued by the SMAP at the LMI.

Additional comments

This service is conceived to be provided instantaneously to the PLME by the PHY, following the PHY's reception of an LM-PH-GET service primitive. The LM-PH-GET service primitives are conceived to be passed across the SAP sequentially. Thus there cannot be multiple outstanding LM-PH-GET requests at the same time, and so there is no need to include, as a parameter of the LM-PH-RETURN, an identifier of the attribute the value of which is returned.

7.3.2.1.3 LM-PH-SET

This service primitive provides an atomic write request service to the PLME.

Semantics of the service primitive

This service primitive is used to change the value of an attribute identified by a registered object identifier to a specified value.

```
LM-PH-SET::= SEQUENCE
                           {
                            [0] AttributeId,
                            [1] AttributeValue
                           }
AttributeId::= PHYAttributeObjId
AttributeValue::= ANY
```

When generated

This service primitive is passed from the PLME to the PHY when an SMAP requires a change of value in a PHY attribute, in accordance with a management application. The SMAP uses the SMI (ISO/IEC 10165-4:1992) defined SET operation at the LMI to command the PLME to set an attribute to a specified value, and the PLME, in turn, passes the LM-PH-SET to the PHY.

Effect of receipt

When the PHY receives this service primitive, it changes the value of the attribute in question. Neither the attribute, identified in the service primitive, nor the ability to change it directly, belongs to the PLME.

Additional comments

The LM-PH-SET is an atomic write operation performed on a single attribute. See Chapter 10 for the use of the SET operation in the specification of the attributes of a PHY MO (or more specifically, an HMUX, PS, or PMD MO).

7.3.2.1.4 LM-PH-ACTION

This service primitive provides a service to the PLME by which it is able to initiate any of a defined set of actions in the already initialized PHY, at the direction of an SMAP application.

Semantics of the service primitive

The parameter of this service primitive is an action descriptor chosen from a list of those actions that the PLME can initiate in the PHY to accomplish layer management tasks.

```
LM-PH-ACTION::= SEQUENCE        {
                                 Action
                                }
Action::= CHOICE   {
                    activate                [0]   NULL
                    deactivate              [1]   NULL
                    reset                   [2]   PHYAttributeSet
                    localLoopback           [3]   NULL
                    remoteLoopback          [4]   NULL
                    provideHtemp            [5]   HTEMP,
                    setHMC                  [6]   NULL,
                    provideSynchPattern     [7]   SynchPattern
                    -- other actions are for further study.
                   }
PHYAttributeSet::= SET OF PHYAttributeAssertion
PHYAttributeAssertion::= SEQUENCE           {
                                             [0]   PHYAttributeObjId
                                             [1]   PHYAttributeValue
                                            }
PHYAttributeValue::= ANY
Htemp::= SEQUENCE OF
                         SEQUENCE
                                            {
                                             [0]   PhSAP,
                                             [1]   Slot-Number,
                                             [2]   Channel-Width OPTIONAL
                                            }
PhSAP::= BIT STRING(SIZE(16))
Slot-Number::= INTEGER (0.2499)
Channel-Width::= INTEGER (0.2499)
SynchPattern::= BIT STRING (SIZE(7))
```

The parameter of this service primitive for the reset action is specified as an unordered list of PHY attribute value assertions, which define values for the attributes to be set, through the use of this service. Any attribute of the PHY, not included in this list, will be set to an already specified default value.

Each item in the unordered list, PHYAttributeSet, is a pair. The first member of the pair is a registered attribute identifier, which is defined as a vector of integers. The second member of the pair is the value to which the attribute is to be set upon initialization.

The Htemp is a complex structure defining the structure of the actual HMUX template, used by the HMUX to assign symbols to slots of the TDM frame structure. The TDM frame structure is conceived of as a sequence of fields of varying width.

When generated

This service primitive is passed from the PLME to the PHY when an SMAP requires an intervention in the operation of the PHY. The SMAP uses one of the SMI (ISO/IEC 10165-4:1992) defined ACTION operations at the LMI to command the PLME to initiate a specific action. The PLME, in turn, passes an LM-PH-ACTION service primitive to the PHY with a parameter corresponding to the ACTION operation requested at the LMI.

Effect of receipt

When the PHY receives the LM-PH-ACTION, it performs the action requested.

The deactivate action causes the affected PHY to suspend functioning so that other SMAP directed actions can be performed upon the PHY of its station's transmission peer (i.e., the AU PHY if the PHY suspended is in the ISTE, or vice versa). The deactivate action might be invoked so that diagnostic procedures could be executed in the PHY of its station's transmission peer.

The activate action causes a PHY to resume functioning if it has been in a suspended state due to having been subject to an earlier deactivate action. If the PHY, which receives the LM-PH-ACTION (activate), is not in a suspended state, then the activate request has no effect upon it.

Upon receipt of the LM-PH-ACTION (reset) service primitive, the PHY resets all of its attributes, using the supplied parameter values and known defaults.

When the PHY of either of a pair of transmission peers receives an LM-PH-ACTION (localLoopback), it proceeds to execute the local loopback requested. When the PHY of the AU receives an LM-PH-ACTION (remoteLoopback), it initiates a remote loopback through the PHY of the ISTE. The LM-PH-ACTION (remoteLoopback) is not defined for the LM-PH service boundary in the ISTE.

The setHMC action is a valueless command which causes the immediate change of an HMUX template, assuming that a provideHtemp action has already been performed.

Additional comments

LM-PH-ACTION (reset) is used when the atomic attribute set request provided above (LM-PH-SET) is not suitable because all or many of the PHY attributes must be set.

In general, actions invoked by LM-PH-ACTION to request services are conceived of as affecting a major part of the operation of the PHY, rather than merely one attribute as is the case with the operations initiated by LM-PH-GET and LM-PH-SET.

7.3.2.1.5 LM-PH-STATUS

This primitive returns to the PLME the result status of an intervention in the PHY initiated by an LM-PH-ACTION being passed to the PHY.

Semantics of the service primitive

The parameters of this service primitive identify the action type and the result status returned by the PHY after an action has been performed.

```
LM-PH-STATUS::= SEQUENCE       {
                                 [0] ActionId,
                                 [1] Status
                               }
ActionId::= ENUMERATED         {
                                 act(0),
                                 deact(1),
                                 res(2),
                                 localLoop(3),
                                 remoteLoop(4),
                                 provideHtemp(5),
                                 setHmc(6),
                                 provideSynch(7)
                                 -- Other values are for further study.
                               }

Status::= ENUMERATED     {
                           success(0),
                           failedNoReason(1)
                           -- Other values are for further study.
                         }
```

When generated

The PHY passes this service primitive to the PLME after having received and satisfied the request of an LM-PH-ACTION from the PLME. In order to maintain conceptual simplicity in the service model, it is assumed that there is no overlapping of actions; no new LM-PH-ACTION is passed by the PLME to the PHY until any outstanding LM-PH-STATUS corresponding to an earlier LM-PH-ACTION has been returned to the PLME. This single instruction stream feature of the service model is not intended to imply any limitation on multithreaded implementations.

Effect of receipt

If the status value returned to the PLME is success, then no further action is defined for this service. If it is anything other than success, then the PHY shall repeat the request for service by passing the associated LM-PH-ACTION to the PHY again, with a maximum of two retries. If, after this number of retries, there is still no success status value returned, an appropriate failure notification is delivered by the PLME to the SMAP, across the LMI.

When the PLME receives an LM-PH-STATUS service primitive, it then completes the satisfaction of an outstanding ACTION service request issued by the SMAP at the LMI.

Additional comments

This service is conceived of as synchronous in the sense that, after passing an LM-PH-ACTION, the PLME remains in a wait state, so far as interactions with the PHY are concerned, until it receives an associated LM-PH-STATUS with the value success. However, the service specification does not depend, in any way, upon the realtime boundaries (i.e., minimum and maximum timer values) of the wait state.

The synchronous feature of the service model is not intended to limit multithreaded implementations.

7.3.2.1.6 LM-PH-NOTIFICATION

This service primitive notifies the PLME of events of management significance that occur during PHY functioning.

Semantics of the service primitive

```
LM-PH-NOTIFICATION::= SEQUENCE       {
                                      Event
                                     }
Event::= ENUMERATED      {
                          hmcParityError(0),
                          tdmParityError(1),
                          lossOfSynch(2),
                          remoteLoopback-Failure(3),
                          localLoopbackFailure(4),
                          pmdInvalid(5),
                          failedAction(6)
                          -- Other values are for further study.
                         }
```

When generated

This service primitive is passed to the PLME by the PHY when one of a defined set of events has occurred in the PHY.

Effect of receipt

When this service primitive is received by the PLME, it may generate a corresponding notification to the SMAP, depending upon the definition of the PHY sublayer MOs. The syntax and the semantics of the specification of the notification to the SMAP at the LMI are defined in Chapter 10 of this standard, in accordance with the SMI (ISO/IEC 10165-4:1992).

Additional comments

This service is conceived of as asynchronous to the operation of the PLME (i.e., the PLME does not enter a prior wait state to receive the notification; rather it is interrupted for this purpose).

8. Detailed specification of the medium access control (MAC) sublayer

8.1 Overview of the MAC sublayer

The MAC sublayer is responsible for establishing, maintaining, and terminating communication between the peer MAC sublayer entities. As described in 5.3.1, the MAC is the lower sublayer in the Data Link Layer (DL). The MAC sublayer provides for the transparent transfer of logical link protocol data units (LPDUs) of a maximum size of 5101 octets to and from the logical link control (LLC) sublayer. Figure 5-4 illustrates the relationship between the MAC sublayer and the other layers. The frame structure of the MAC frame is illustrated in figure 6-14.

The integrated services local area network (ISLAN) MAC sublayer has been described as a "protocol engine" consisting of several separate functional processes. These processes are:

a) *MAC state control.* Governs overall coordination of all MAC processes.
b) *MSDU control.* Governs the send/receive of MAC service data units (MSDUs) to/from the upper layer processes.
c) *LM message control.* Governs request/notification messages to/from layer management.
d) *Access control (AC).* Governs the request/grant protocol through which the ISTE and the access unit (AU) gain access to the P channel for the purpose of transport of a MAC protocol data unit (MPDU).
e) *MPDU control.* Governs bit stream presentation of the MPDU to the hybrid multiplexer (HMUX) sublayer and bit stream reception of the MPDU from the HMUX sublayer.
f) *Resource control.* Governs allocation of buffer resources for all of the other MAC sublayer functional modules.
g) *Timer control.* Governs timer administration for all of the other MAC sublayer functional modules.

8.2 The ISLAN MAC protocol engine model

The sequence and description language (SDL) diagram of the ISLAN MAC protocol engine, presented in figure 8-1, defines a model of the internal architecture of the MAC protocol processing block. Each of the octagonal figures represents a concurrent process within the block. The differentiation of such processes provides a logical decomposition of the ISLAN MAC architecture. These processes are conceived to interact either asynchronously or synchronously by the exchange of messages, called "signals" in SDL. These signals carry information content between processes and between the internal processes and various external protocol blocks.

The SDL diagram, in figure 8-1, conforms to CCITT Recommendation Z.100 (1993), except for a few small diagrammatic deviations, such as the explicit representation of intraprocess queues. This model is intended to be conceptual in nature; it does not restrict the range of implementation architectures.

The following subclauses provide a functional description of the processes.

8.2.1 MSDU_CNTRL process description

This process exchanges the MA-UNITDATA request and indication service primitives [i.e., the ma_ud_req() and the ma_ud_ind() signals] with the MAC service user. Each signal carries an MSDU, as a parameter, in one direction or the other. The signal exchange takes place across the bidirectional MSDU_C channel.

After receiving an MSDU in an MA-UNITDATA request [i.e., the ma_ud_req() signal], MSDU_CNTRL creates the corresponding MPDU and sends it, as a parameter of a pdu_to_send() signal, to MAC_STATE_CNTRL. This signal sequence is conceived of as immediate; no MPDU queue is maintained in MSDU_CNTRL.

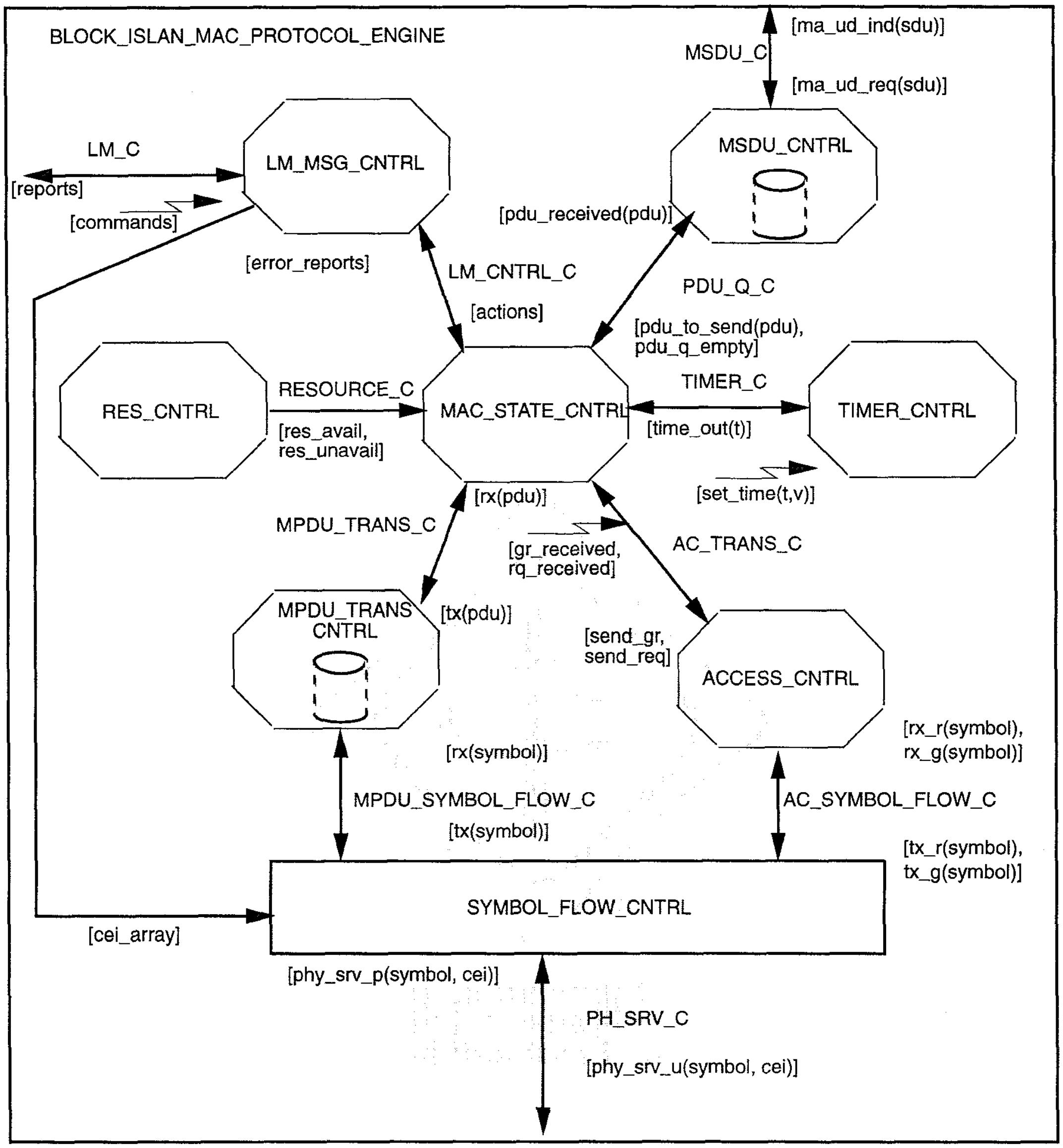


Figure 8-1—ISLAN MAC protocol engine SDL diagram

After receiving an MPDU from MAC_STATE_CNTRL, as the parameter of a pdu_received() signal, MSDU_CNTRL creates the corresponding MSDU, by extracting the data field from the MPDU, and sends the MSDU, as a parameter of an ma_ud_ind() signal, to the MAC service user.

The signal exchange between MSDU_CNTRL and MAC_STATE_CNTRL takes place across the bidirectional PDU_Q_C channel.

8.2.2 LM_MSG_CNTRL process description

This process provides an intermediary between the layer management entity (LME), representing the system management application process (SMAP), both of which are external to the current block, and the control of the internal states of the ISLAN_MAC_PROTOCOL_ENGINE. The LM_MSG_CNTRL process is basically a message handler between the manager and the MO.

From the LME (and ultimately the external SMAP), LM_MSG_CNTRL receives layer management commands. It also returns reports to the LME (and the SMAP), including the following:

a) Attribute values requested in the layer management commands.
b) Reports on the results of layer management actions initiated by SMAP commands originated from outside the ISLAN_MAC_PROTOCOL_ENGINE block.
c) Unsolicited, asynchronous notifications to the LME (and the SMAP). This signal exchange of commands and reports takes place across the bidirectional LM_C channel.

As the foregoing description indicates, "commands" and "reports," in figure 8-1, are identifiers for SDL signal lists, the constituents of which are not specified here in detail.

Corresponding to the commands and reports flowing between LM_MSG_CNTRL and the SMAP, there are error_report and action signals flowing between LM_MSG_CNTRL and MAC_STATE_CNTRL, across the bidirectional LM_CNTRL_C channel.

The action signals cause MAC_STATE_CNTRL to initiate functions which correspond to the management semantics of the SMAP commands.

"Error_reports" and "actions" are identifiers for SDL signal lists, the constituents of which are not specified here in detail.

8.2.3 MAC_STATE_CNTRL process description

This process exercises the central control functions of the request/grant protocol. It effectively coordinates all of the other concurrent processes in the ISLAN_MAC_PROTOCOL_ENGINE block, so far as the request/grant protocol is concerned.

The finite state machines (FSMs) described in 8.3 define the state changes of the MAC_STATE_CNTRL process.

The MAC_STATE_CNTRL process controls an internal queue of MPDUs, i.e., those which have been passed from the MSDU_CNTRL process for transmission outbound to the transmission peer [i.e., terminal equipment (TE) MAC or access unit (AU) MAC], subject to the request/grant protocol.

This outbound MPDU queue is one of two internal queues within the ISLAN_MAC_PROTOCOL_ENGINE block required by the request/grant protocol. The second is an inbound MPDU queue maintained in the MPDU_TRANS_CNTRL process (see 8.2.6).

8.2.4 RES_CNTRL process description

This process monitors the availability of internal processing resources required for the sending and receiving of MPDUs. The primary resource in question is a set of buffers for the transient holding of one or more MPDUs in the internal queues controlled by the MAC_STATE_CNTRL process and the MPDU_TRANS_CNTRL process, respectively.

Whenever there is a change between the availability and the unavailability of this type of resource for the queue of incoming MPDUs controlled by MPDU_TRANS_CNTRL, this change is indicated to MAC_-STATE_CNTRL by RES_CNTRL sending it either a res_avail or a res_unavail signal. In regard to buffer space, receipt of res_avail means that there is enough buffer space so that at least one incoming MPDU can be buffered in MPDU_TRANS_CNTRL until it can be passed to MAC_STATE_CNTRL. Receipt of res_unavail means that there is no longer at least this much buffer space available for the incoming queue controlled by MPDU_TRANS_CNTRL.

The availability of buffer space in the inbound queue that is controlled by MPDU_TRANS_CNTRL is one of the primary determinants of whether a GRANT=YES will be sent to the transmission peer. Availability of space in the inbound queue may indirectly be a function of the amount of buffer space allocated for the other internal queues; however, this functional effect is implementation dependent.

The res_avail and the res_unavail signals are sent across the unidirectional (RES_CNTRL==>MAC_-STATE_CNTRL) RESOURCE_C channel.

8.2.5 TIMER_CNTRL process description

This process controls all internal timers required for the operation of the request/grant protocol and related functions within the ISLAN_MAC_PROTOCOL_ENGINE. No timer controlled by TIMER_CNTRL provides any function other than countdown.

A timer is set by a request from MAC_STATE_CNTRL, and is sent as the set_time(t,v) signal, where t designates a specific timer and v is a nonnegative integer which is requested to be the countdown value (i.e., the starting number of ticks) of the timer.

The time-out event of a specific timer, t, is indicated to the MAC_STATE_CNTRL process by its receipt of a time_out(t) signal from TIMER_CNTRL.

The set_time() and the time_out() signals are passed between MAC_STATE_CNTRL and TIMER_CNTRL across the bidirectional TIMER_C channel.

8.2.6 MPDU_TRANS_CNTRL process description

This process controls the serialization of an MPDU for outbound transmission to the corresponding transmission peer and the reverse function of assembling an inbound MPDU from a symbol stream.

In order to assemble an MPDU from an inbound symbol stream, buffer space is required for concatenation of the octets that constitute the MPDU. The availability of this buffer space is a necessary and sufficient condition for the sending of a GRANT=YES. This buffer space is also referred to above as an inbound MPDU queue.

The MPDU_TRANS_CNTRL process has two signalling channel interfaces:

a) The MAC_STATE_CNTRL process
b) The SYMBOL_FLOW_CNTRL block

Across item a) the bidirectional MPDU_TRANS_C channel between MAC_STATE_CNTRL and MPDU_TRANS_CNTRL, signals carry inbound MPDUs [i.e., as the parameters of the rx() signals] and outbound MPDUs [i.e., as the parameters of the tx() signals].

Across item b) the bidirectional MPDU_SYMBOL_FLOW_C channel between MPDU_TRANS_CNTRL and SYMBOL_FLOW_CNTRL, signals carry inbound symbols [i.e., as the parameters of rx() signals] and outbound signals [i.e., as the parameters of the tx() symbols]. Across this signalling interface MPDU_TRANS_CNTRL always responds synchronously to the receipt of an rx(symbol) signal by immediately passing a tx(symbol) signal to SYMBOL_FLOW_CNTRL. In order to make asynchronous transmission of MPDUs possible, the symbol may have a value of the Idle-Symbol whenever there are no data symbols to be transmitted because there is no MPDU to be serialized. This synchronization corresponds to the synchronous PHY service, explained in 7.1.5.4.1.

8.2.7 ACCESS_CNTRL process description

This process controls the encoding, in symbols, of requests for permission to transmit an MPDU, and grants of this permission.

The ACCESS_CNTRL process has two signalling channel interfaces:

a) The MAC_STATE_CNTRL process
b) The SYMBOL_FLOW_CNTRL block

Across step a) the bidirectional AC_TRANS_C channel between ACCESS_CNTRL and MAC_STATE_-CNTRL,

1) A signal (i.e., send_gr) indicates that permission is to be sent to the transmission peer for it to send an MPDU
2) A signal (i.e., send_req) indicates that a request is to be sent to the transmission peer to be allowed to send an MPDU to that peer
3) A signal (i.e., gr_received) indicates that a grant of permission to send an MPDU to the transmission peer has been received from the transmission peer
4) A signal (i.e., rq_received) indicates that a request for permission has been received from the transmission peer for it to send an MPDU

Across step b) the bidirectional AC_SYMBOL_FLOW_C channel between ACCESS_CNTRL and SYMBOL_FLOW_CNTRL, signals (i.e., rx_r(symbol) and rx_g(symbol)) are passed carrying, as parameters, request and grant symbol values inbound from the transmission peer. Across the same channel interface, signals (i.e., tx_r(symbol) and tx_g(symbol)) are passed carrying, as parameters, request and grant symbol values outbound to the transmission peer.

Request and grant symbol values must be either Binary-1 (encoding YES); or Binary-0 (encoding NO). The Idle-Symbol cannot be used.

8.2.8 SYMBOL_FLOW_CNTRL process description

SYMBOL_FLOW_CNTRL is not a process, but an SDL block, the internal structure of which will not be presented here.

However, the general function of this block is to maintain the correspondence between a symbol flow and a PHY Connection Endpoint Identifier (CEI). Outside the ISLAN_MAC_PROTOCOL_ENGINE block, each CEI is mapped into a specific PhSAP, but the distinctions between the PhSAPs and their correlation to PHY channels known at the HMUX sublayer should be unknown to the internal structures and FSMs of the ISLAN_MAC_PROTOCOL_ENGINE block. This is a requirement of proper protocol layering, and it is a requirement that is supported by the use of CEIs.

As explained above, symbol exchanges go on between SYMBOL_FLOW_CNTRL, on the one hand, and MPDU_TRANS_CNTRL and ACCESS_CNTRL, on the other. Associations of a symbol and a CEI are exchanged across the bidirectional PH_SRV_C channel, using the phy_srv_p() signal (passed from the PHY service provider) and the phy_srv_u() signal (passed from the PHY service user).

LM_MSG_CNTRL provides an array of CEIs for the symbol flows, using the cei_array signal.

8.3 P channel access control

This standard defines a packet channel, the P channel, through which each ISTE has access to the LAN services. This clause describes the access control method by which the AU and the ISTE request access to the P channel for the purpose of transport of an MPDU.

8.3.1 Overview

The IEEE 802.9 interface defines a point-to-point full-duplex channel, a P channel, between each ISTE and the AU to which it attaches. The bandwidth of this P channel may vary among ISTEs in service. Figure 8-2 illustrates a sample configuration in which "n" ISTEs are attached to an AU where Pi indicates the P channel between the i-th ISTE and the AU. In general, the bandwidth of the P channel, Pi, varies and depends on how much of the interface bandwidth has been allocated to the isochronous C channel(s) and B channel(s) in the given 802.9 channel between the ISTE and the AU. A sample configuration of multiple ISTEs connected to an AU is illustrated in figure 8-2.

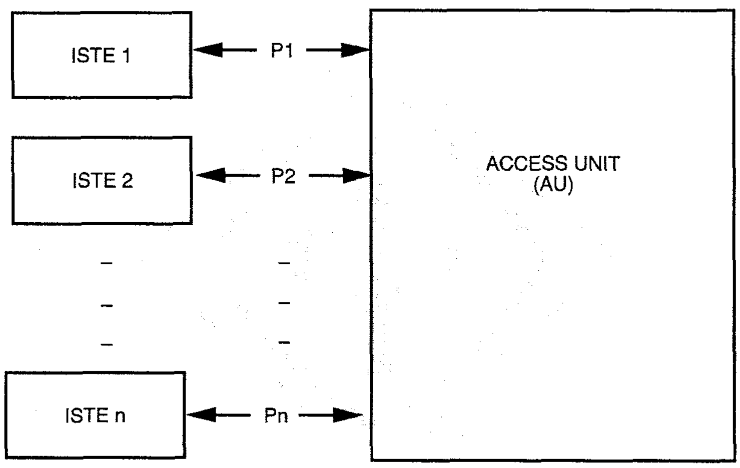


Figure 8-2—Sample configuration of ISTEs connected to the AU

The usage of each P channel is governed by the ISTE and the AU using a control protocol known as the request/grant protocol.

8.3.1.1 Characteristics of request/grant protocol

The following is a summary of the characteristics of the request/grant protocol procedure.

a) The request/grant protocol is associated with the transmission and reception of MPDUs.

b) The transmission of a request signal set to "1" (R=1) requests permission for the station to send one complete MPDU.

c) The transmission of a grant signal set to "1" (G=1) indicates that receive resources are available to receive one MPDU.

d) The reception of a grant signal set to "1" (G=1) allows the station to send one complete MPDU.

e) The transmission of MPDUs from an ISTE to the AU ("upstream" transmission) is governed and controlled by the AU.

f) The transmission of MPDUs from the AU to an ISTE ("downstream" transmission) may or may not be governed and controlled by the ISTE. If the AU's management entity has set the IGNORE_GRANT variable to "1," then the AU shall not await for a Grant=1 signal from an ISTE before transmitting an MPDU to it.

8.3.1.2 Definitions of internal signals, external signals, and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this AC method. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

8.3.1.2.1 External signals

The following primitive signals appear as external stimulus to the request/grant FSM and the transmit/receive FSM:

LM-MA-SET(IGNORE_GRANT)	This is a management request. If possible, it will set IGNORE_GRANT to the indicated state, "1" or "0."
LM-MA-ACTION(activate)	When this management request is issued, the MAC state machine will initialize all state variables and timers. Upon the completion of this task, the MAC state machine shall enter its operational states.
LM-MA-ACTION(deactivate)	When this management request is issued, the MAC state machine will stop all functional processing and return to an inactive state.
LM-MA-ACTION(reset)	When this management request is issued, the MAC state machine will reset all state variables and timers to the default values. Upon the completion of this task, the MAC state machine shall return to its normal functional state.

8.3.1.2.2 Internal signals

The following internal signals are generated by the functional processes within the request/grant FSM and the transmit/receive FSM. They are commonly referred to as "flags" and "event/error counters."

DECREMENT_RCV_QUEUE	This internal signal is used to remove one resource from the pool of receive buffers in the MPDU receive control queue. This results in the formation of the RX(pdu) signal that is sent up to the MAC state control module.
DECREMENT_XMT_QUEUE	This internal signal is used to remove one PDU (which is queued) that is awaiting transmission in the MPDU transmit control queue.
GRANT_BIT	This signal bit appears in the AC field of the TDM frame. This bit carries the grant signals. This is set to "1" by the granter/receiver whenever it is ready to receive an MPDU. Note by definition, the receive buffer shall be equal to the maximum size of the MPDU plus a reserve buffer that has sufficient capacity to receive one TDM frame payload field. The size of this reserve buffer may be set so as to accommodate part of the payload that is delivered between AC fields for the situation in which the aggregate stream has delivered its payload field in a multiple row format.
GRANT_RECVD	This is a Boolean status indicator. This variable is set to value "1" if the state of the last received GRANT_BIT was set to "1." Otherwise it is set to "0."

IGNORE_GRANT	This is a Boolean status indicator. If this value is "1," the AU or ISTE may send an MPDU without receiving a GRANT_BIT="1." The design of a node may be such that this bit is always "1." Note that the AU management entity shall set this flag to "1" as the default operational mode. The ISTE management entity shall set this flag to "0."
PDU_TO_SEND	This is a Boolean status indicator. This indicates whether or not there are one or more MPDUs ready for transmission (excluding the one being transmitted). It is set to "1" if there is at least one complete MPDU ready for transmission in the queue that exists between the MSDU control module and the MAC state control module. Otherwise, this variable is set to "0."
PDU_BEING_SENT	This is a Boolean status indicator. This indicates whether or not an MPDU is currently being sent/transmitted by the MPDU transmit control module. It is set to "1" if an MPDU is being sent. Otherwise, this variable is set to "0."
PDU_BEING_RECEIVED	This is a Boolean status indicator. This indicates whether or not an MPDU is currently being received by the MPDU receive control module. It is set to "1" if an MPDU is being received. Otherwise, this variable is set to "0."
REQUEST_BIT	This is a signal that appears in the AC field of the TDM frame. This bit carries the request signals.
RX_EOP	This is a binary indicator. The value "1" indicates the receive end of PDU.
RX_INDICATION(pdu)	This is a Boolean status variable that indicates the arrival of an incoming MPDU.
RX_REQUEST	This is a Boolean status variable used to store the most recently received request signal. This variable is set to "1" when REQUEST_BIT="1" has been received. Otherwise this variable is set to "0."
RX_RES_AVAIL	This is a binary indicator. The value "1" indicates a receive resource is available.
RX_SOP	This is a binary indicator. The value "1" indicates the receive start of PDU.
TX_GRANT	This is a Boolean status variable used to store the transmitted GRANT_BIT status.
TX_PDU_TO_SEND	This is a Boolean status indicator. This indicates whether or not there are one or more MPDUs ready for transmission. It is set to "1" if there is at least one complete MPDU ready for transmission in the queue that exists between the MAC state control module and the MPDU transmit control module.

8.3.1.2.3 State names for request/grant FSM

The following are definitions of the states used in the functional model for the request/grant FSM.

The request/grant transmitter FSM states are as follows:

AWAIT_MPDU	The AWAIT_MPDU state is reached upon activation by the LME. In this state, the transmitter FSM is awaiting work requests from the MAC entity to send PDUs.
INACTIVE	The INACTIVE state is the deactivated condition. In this state, the transmitter FSM will not respond to any stimulus except for messages from its LME.
REQUEST	The REQUEST state is the state in which the transmitter FSM sends a REQUEST_BIT=1 to the peer entity in accordance with the REQUEST_TIME_SLOT indication from the PHY Layer, because there is at least one MPDU queued to be sent. The transmitter FSM shall stay in this state and repeat sending REQUEST_BIT=1 until the variable GRANT_RECVD turns to "1" or until the transmit request timer (TX_REQ_TIMER) expires. When GRANT_RECVD turns to "1," the MPDU queued to be sent is set in the process of start sending in the P channel payload of the TDM frame. Upon the occurrence of this event, the transmitter FSM will vector back to the AWAIT_PDU state so that another request cycle may be repeated. In the event of TX_REQ_TIMER EXPIRED, the FSM will notify layer management of the error event.

The request/grant receiver FSM states are as follows:

AWAIT_RES	The AWAIT_RES state is reached upon activation by the LME. In this state, the receiver FSM is awaiting availability of resources in order that it may send a GRANT to the peer entity.
INACTIVE	The INACTIVE state is the deactivated condition. In this state, the receiver FSM will not respond to any stimulus except for messages from its LME.
GRANT	THE GRANT state is the state in which the receiver FSM sends a GRANT_BIT=1 to the selected peer entity in accordance with the GRANT_TIMER_SLOT indication from the PHY Layer, because there is a resource available to receive at least one MPDU plus one P channel payload equivalent buffer. The receiver FSM shall stay in this state and repeat sending GRANT_BIT=1 until RX_INDICATION(PDU) turns to "1" or until receive grant timer (RX_GRANT_TIMER) expires. Upon the occurrence of these events, the receiver FSM will vector back to the AWAIT_RES state so that another grant cycle may be repeated. In the event of plural RX_GRANT_TIMER_EXPIRED, the FSM will notify layer management of the error event.

8.3.1.2.4 State names

The following are definitions of the states used in the functional model for the MPDU transmit/receive FSM.

The MPDU transmitter FSM states are as follows:

IDLE	The IDLE state is reached upon activation by the LME. In this state, the transmitter FSM is awaiting work requests from the MAC entity to send PDUs.
INACTIVE	The INACTIVE state is the deactivated condition. In this state, the transmitter FSM will not respond to any stimulus except for messages from its LME.
SEND	The SEND state is the state in which the transmitter FSM sends an MPDU. The transmitter FSM shall stay in this state until after the last bit of the PDU has been sent. Upon the occurrence of this event, the transmitter FSM will vector back to the IDLE state so that another transmit cycle may be repeated.

The MPDU receiver FSM states are as follows:

IDLE	The IDLE state is reached upon activation by the LME. In this state, the receiver FSM is awaiting the receipt of a start of incoming packet (RX_SOP=1) indication, coupled with the availability of a receive buffer. Once this event occurs, the FSM will vector to the "RECEIVE" state and begin the process of receiving an incoming PDU.
INACTIVE	The INACTIVE state is the deactivated condition. In this state, the receiver FSM will not respond to any stimulus except for messages from its LME.
RECEIVE	The RECEIVE state is the state in which the receiver FSM will begin to store incoming bits of information until an end of PDU (RX_EOP=1) indication is received. The MPDU receive control FSM will be responsible for counting incoming bytes. Following the complete reception of an incoming PDU, the FSM will be responsible for conducting minimum/maximum PDU error checking. Upon detection of a complete incoming PDU, the MPDU receive control FSM will vector back to the IDLE state so that another receive cycle may be repeated.

8.3.1.2.5 Managed objects

The following objects within the FSM are controlled or "managed" by the management entity within the ISTE and AU:

max_mpdu_size_received	This managed variable indicates the maximum length of MAC frame that will be received. The maximum size MAC frame allowed is 5100 octets (see 6.5.11.2).
min_mpdu_size	This integer is the minimum number of octets in an MPDU. The minimum size MAC frame allowed is 19 octets (LEN, FC, DA, SA, FCS). See 6.5.11.2.

rx_octet_count	This is an integer count of the number of octets received in an MPDU.
rx_grant_timer	This managed 16-bit timer indicates how long a grant will be asserted without having received a start of packet indication.
rx_gtimer_errcnt	This managed 16-bit counter is incremented each time a REQUEST GRANT timer has reached its expiry condition.
rx_max_errcnt	This managed counter is incremented if a received MPDU length exceeds the maximum transmitted unit length.
rx_max_err_cnt_thresh	This is a threshold counter for management action.
rx_min_errcnt	This managed integer counter is incremented if a received MPDU is received that is undersize.
rx_res_errcnt	This managed 16-bit counter is incremented on the condition that a received start of packet has occurred but no receive buffer resources are available.
tx_req_timer	This managed 16-bit timer indicates how long a request will be asserted without having received a grant.
tx_req_errcnt	This managed 16-bit counter is incremented when a grant is not received while the TX_REQ timer has expired "n" times.

8.3.1.2.6 Request/grant procedures and timing diagram

The AC field of the TDM frame consists of three request bits: REQ3, REQ2, and REQ1 (figure 6-7). These bits supports multiple priority requests. The use of multiple priorities of requests is implementation dependent. It is assumed that higher priority MAC frames (MPDUs) are processed prior to lower priority MAC frames. Priority 3 is the highest. When priority 3 MPDU requires service, REQ3 is sent for a grant; when priority 2 MPDU requires service, REQ2 is sent; and for priority 1 (the lowest priority) MPDU, REQ1 is sent.

When multiple ISTEs connected to an AU send requests of different priorities, the AU will provide grant to the highest priority request first as part of resource allocation and flow control. The nature of the resolution algorithm for flow control and resource allocation is dependent on the AU implementation. The normal operation mechanism is for the ISTE to only set one priority request at a time. The procedures for simultaneously issuing multiple request priorities at the ISTE is to only set one priority request at a time.

The request (R) and grant (G) subfields of the AC field of the TDM frame are used to synchronize the MAC frame ready for transport between the ISTE and the AU.

The following definitions apply to the use of the request (R) and grant (G) signals:

R=0	No request outstanding for MPDU transmission.
R=1	Request outstanding for pending MPDU transmission.
G=0	No resource is available in the 802.9 device (either AU or ISTE) to receive an MPDU.
G=1	Resource is available for receiving an MPDU in the AU. The G=1 is issued by the receiver (either AU or ISTE) whenever it has a resource available to receive this MPDU. This action is independent of the request that was issued from the peer device.

The resource must be equal to the combination of one maximum length MPDU plus one TDM frame's payload. Allocating this resource size is also appropriate for the situation in which multiple short length MPDUs are to be sent, since these packets could be received in successive TDM frames without the need for request/grant protocol exchange introducing delay in the pending transmission.

RES — This is a receive storage resource that can accommodate one MPDU. Whenever the AU issues a G=1, it has allocated a receive resource that has a length equal to the maximum MPDU size plus one reserve buffer equal to one TDM frame size. The AU expects to receive one MPDU (of length less than or equal to one maximum MPDU).

The G=1 is sent by the AU to the ISTE if it has the resources to receive one MPDU plus one TDM frame. Whenever the ISTE has an MPDU ready for transport, it will assert R=1 in the next available TDM frame and await a G=1 from the AU in the return TDM frame. However, if G=1 is received while the MPDU is ready for transport, the ISTE will immediately initiate transport of the MAC frame without sending R=1. Once G=1 is received, the ISTE will send the entire MAC frame to the AU. This MAC frame will be sent in the P channel of consecutive TDM frames.

The start of frame (SOF) is set to "1" in the TDM frame period wherein the first octets of the MPDU pending transmission are sent. The SOF field is interpreted in the request/grant procedure for distinguishing different grants (e.g., G1, G2, etc.) from the AU as follows:

a) The G=1 received in successive incoming TDM frames is to be interpreted as the grant for the pending MPDU.
b) If the MPDU transmission has begun in a TDM frame, the SOF is set to "1" in the AC field of that frame.

The G=1 which is received in the return TDM frame (after the SOF=1 frame was transmitted) is interpreted as a grant for the next MPDU pending transmission. This handshaking provides stability in handling grant errors and in coordinating the realtime availability of the buffer resources in the AU. Thus the use of the SOF provides for a robust request/grant protocol.

If the AU has a MAC frame ready for transport to the ISTE, it will send the frame whenever it has the TDM frame's P channel slot available. From a network management perspective, the AU will not wait for G=1 from the ISTE prior to transmission of the MAC frame. In other words, G=1 is always asserted (in the AC field of the TDM frame) for the direction of frame transmission from the ISTE toward the AU.

When multiple ISTEs are connected to a single AU and resources are commonly shared, the requests from each of these ISTEs are used by the AU for resolving contention for resources. The contention resolution procedure within the AU is implementation dependent.

The following examples illustrate the request/grant protocol with respect to MAC frame transport from the ISTE to the AU. The examples illustrate two cases:

— The single request case
— The multiple request case

The timing relationship between the TDM frame transmissions exchanged between the ISTE and the AU is dependent on the IEEE 802.9 network environment. A detailed description is presented in Chapter 9.

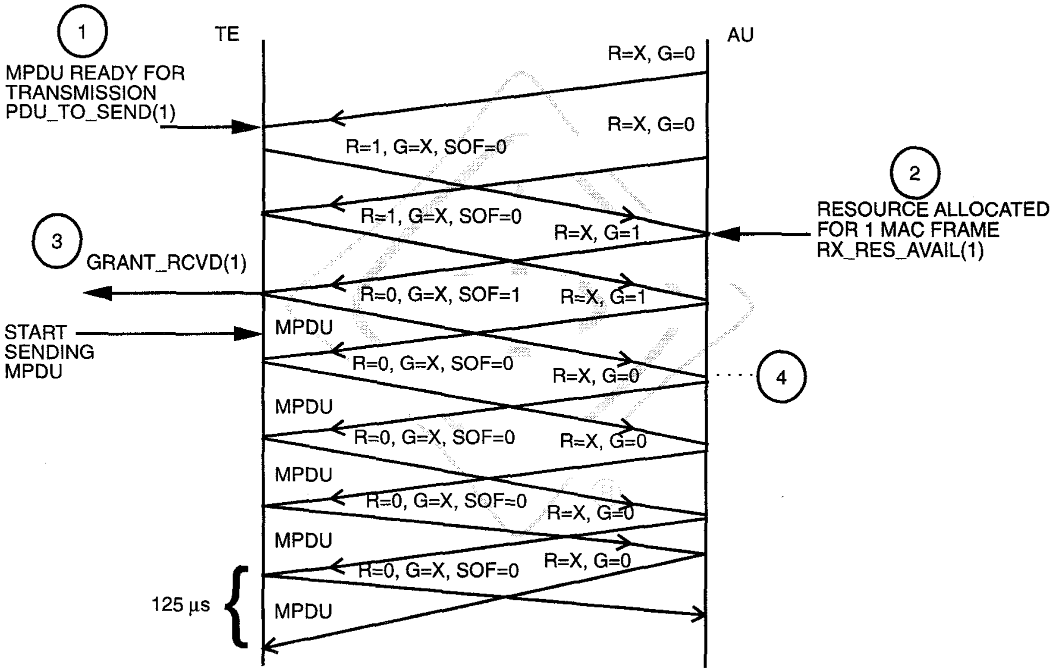
8.3.1.2.6.1 Single request case

In this situation, the ISTE has a single MPDU queued to be sent to the AU. The ISTE gains access to the physical medium in two possible sequences using the request/grant protocol:

a) MPDU is ready to be sent and G=0 is asserted by the AU
b) MPDU is ready to be sent and G=1 is asserted by the AU

Single request case with initial grant = 0

Figure 8-3 illustrates the example of a single request that is queued in the ISTE and initially G=0 has been received from the AU. When the MPDU is ready for transmission at the ISTE, R=1 is sent by the ISTE in the AC field of the next available TDM frame. The ISTE awaits the G=1 from the AU in order to send the MPDU.



NOTES:
R: REQUEST BIT IN TDM FRAME
R=1 FOR REQUEST
R=0 FOR NO REQUEST
R=X FOR DON'T CARE

G: GRANT BIT IN TDM FRAME
G=1 FOR GRANT
G=0 FOR NO GRANT
G=X FOR DON'T CARE

Figure 8-3—Single request timing diagram with initial grant state, G=0

The steps for exchange of request/grant information are as follows:

a) When an MPDU is ready for transmission at the ISTE, the request bit is set to "1" (R=1) in the next AC field that is transmitted from ISTE toward AU. This request is received at the AU within 125 µs.

b) In the next AC frame period, the AU asserts the grant bit set to "1" (G=1) if resources are available to receive the MPDU (of maximum length).

c) Since no further MPDUs are ready for transport, the ISTE will assert R=0 and will proceed to "construct" the TDM frame by setting SOF=1 in the AC field of the pending TDM frame and then proceeding to transmit the TDM frame.

d) Since the permission to send was only for one MPDU, the AU will immediately assert G=0 and send this TDM frame to the ISTE.

Single request case with initial grant = 1

Figure 8-4 illustrates the example of a single request that is queued in the ISTE and initially G=1 has been received from the AU (i.e., the AU has resources available to receive one MPDU).

The ISTE will begin transmission of the MPDU in the next available TDM frame and will set SOF=1 in that frame. Since only one MPDU is queued for transmission (in this example), the ISTE will immediately set R=0 in the next TDM frame period, since it has already received permission to send one MPDU. If the MPDU transmission continues in the subsequent TDM frame periods, then the ISTE will continue to send R=0 and SOF=0.

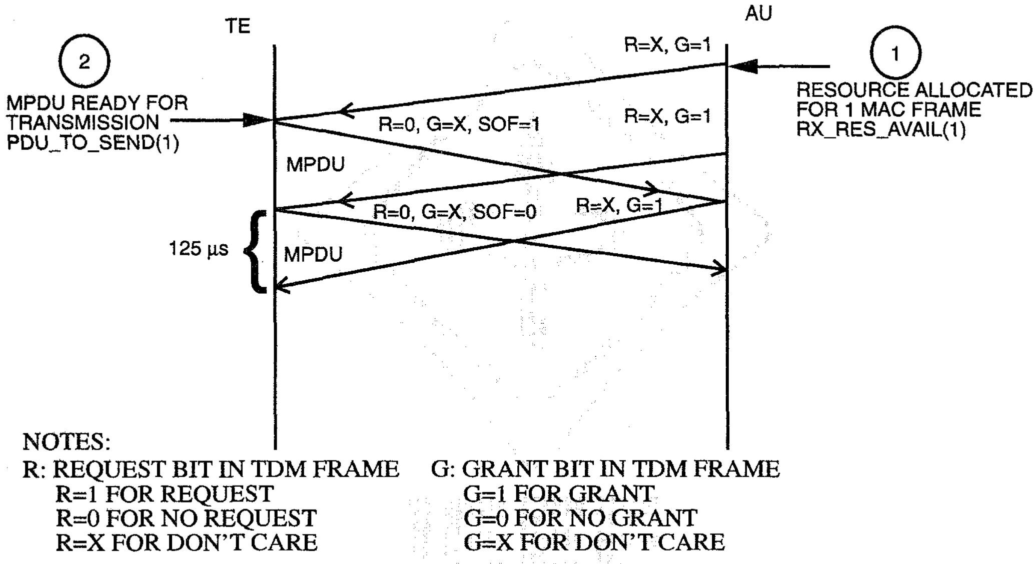


Figure 8-4—Single request timing diagram with initial grant state, G=1

8.3.1.2.6.2 Multiple request case

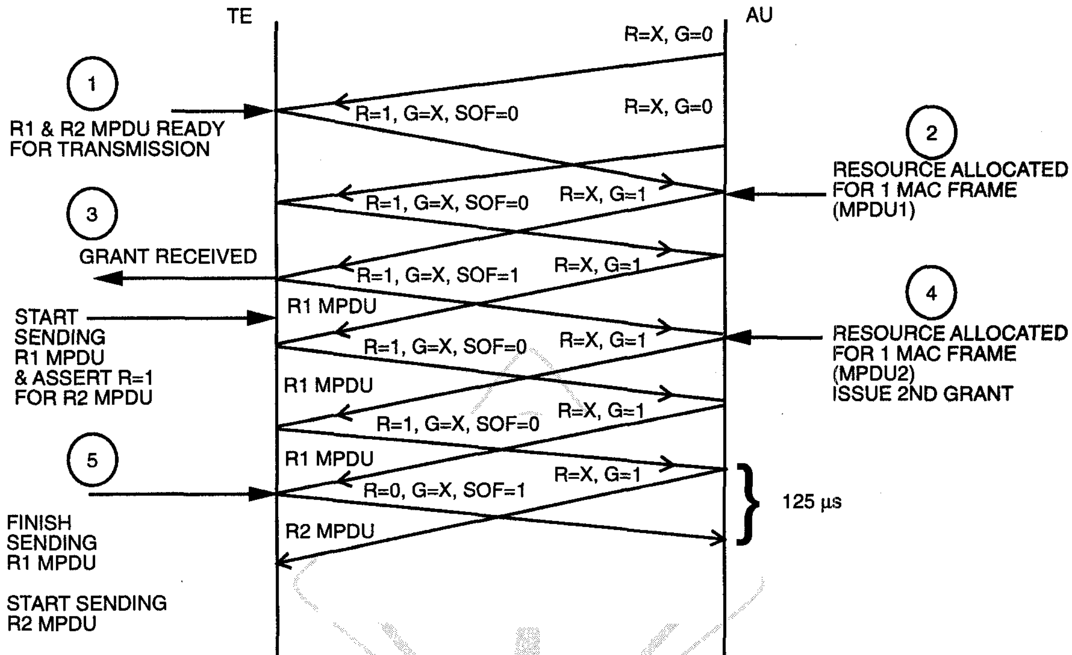
When more than one MPDU is readied at the ISTE by different applications, there will be more than one request that needs to be serviced. These requests will be arbitrated within the ISTE and a particular request will be processed.

The ISTE gains access to the physical medium in two possible sequences using the request/grant protocol:

a) MPDUs are ready to be sent and G=0 is asserted by the AU
b) MPDUs are ready to be sent and G=1 is asserted by the AU

Multiple request case with initial grant = 0

Figure 8-5 illustrates the example of multiple requests that are queued in the ISTE and initially G=0 has been received from the AU. When the MPDU is ready for transmission at the ISTE, R=1 is sent by the ISTE in the AC field of the next available TDM frame. The ISTE awaits the G=1 from the AU in order to send the MPDU.



NOTES:

R: REQUEST BIT IN TDM FRAME
R=1 FOR REQUEST
R=0 FOR NO REQUEST
R=X FOR DON'T CARE

G: GRANT BIT IN TDM FRAME
G=1 FOR GRANT
G=0 FOR NO GRANT
G=X FOR DON'T CARE

Figure 8-5—Multiple request timing diagram with initial grant state, G=0

The steps for exchange of request/grant information are as follows:

a) When MPDUs are ready for transmission at the ISTE, the request bit is set to "1" (R=1) in the next TDM frame that is transmitted from the ISTE toward the AU. This request is received at the AU within 125 µs. The ISTE continues to send R=1 in subsequent TDM frames until G=1 is received from the AU.

b) In the next TDM frame period, the AU sends the grant bit set to "1" (G=1) if resources are available to receive one MAC frame (MPDU1). The AU will maintain G=1 in subsequent TDM frames until it receives an incoming TDM frame which contains SOF=1 in the received AC field. The status of SOF=1 designates that this received TDM frame's payload field contains the start of MPDU1.

c) Upon reception of G=1 for the MPDU1 that is queued for transmission, the ISTE will start transmitting MPDU1 in the very next available TDM frame. At this current moment, the SOF field will be set to "1," and the R bit will be set to "1" to designate that there is another MPDU pending for transmission (MPDU2). If the MPDU (MPDU1) requires multiple payload periods to be sent, the ISTE will continue to send R=1 and SOF=0 in each delivered frame.

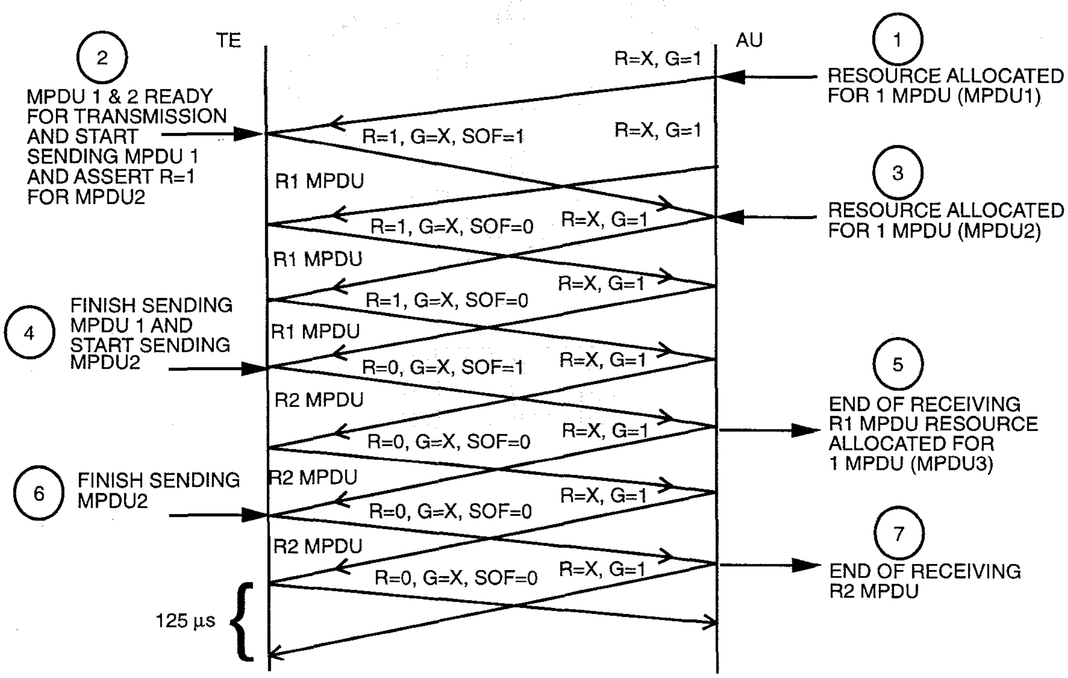
d) Upon reception of SOF=1 from the incoming TDM frame, the AU will detect R=1 which requests another available resource. If the resource is available to receive a second MPDU (MPDU2), then the AU will send a TDM frame with G=1. The AU will continue to send G=1 in subsequent TDM frames to acknowledge the request for MPDU2 reception until it receives another TDM payload unit with the SOF=1.

e) Once the MPDU1 frame has been completely transmitted, the ISTE will begin sending MPDU2 since it has already received the grant permission. Note that in the first payload component of MPDU2, ISTE will set SOF=1. If MPDU2 is continued to be transmitted across subsequent TDM frames, the SOF bit will be set to "0" in these frames.

Multiple request case with initial grant = 1

This case assumes that there is only one AC field per TDM frame and only one request priority is used. Since the ISTE has more than one MPDU ready for transmission, multiple requests are required to be sent to the AU.

Figure 8-6 illustrates the example of multiple requests that are queued in the ISTE and initially G=1. Since G=1 has been received, the ISTE will begin transmission of the first MPDU (MPDU1) in the next available TDM frame and will set SOF=1 to indicate that the frame transmission has started. Also, since MPDU2 is ready for transmission, the ISTE will send R=1 in the same TDM frame. While MPDU1 is continued to be transmitted in subsequent TDM frames, R=1 is still maintained. However, SOF=0 will be sent in these TDM frames.



NOTES:
R: REQUEST BIT IN TDM FRAME
R=1 FOR REQUEST
R=0 FOR NO REQUEST
R=X FOR DON'T CARE

G: GRANT BIT IN TDM FRAME
G=1 FOR GRANT
G=0 FOR NO GRANT
G=X FOR DON'T CARE

Figure 8-6—Multiple request timing diagram with initial grant state, G=1

The AU may issue G=1 in subsequent frames for the next MPDU transmission. However, the ISTE will send MPDU2 after it has completely transmitted MPDU1. The AU continues to send G=1 in the TDM frames for MPDU2 reception until the ISTE begins sending MPDU2 with SOF=1 in a TDM frame. As long as there is a resource available in the AU to receive another MPDU, the AU will continue to send G=1 in subsequent TDM frames. If no more MPDUs are pending for transmission, the ISTE will send R=0 in subsequent frames.

The steps for exchange of request/grant information are as follows:

a) Since the AU has resources available for receiving an MPDU, it will send G=1 in the next available TDM frame to the ISTE. This grant is received at the ISTE in approximately 125 μs.

b) The ISTE shall start transmitting the MPDU1 in the next available TDM payload frame and shall set SOF=1 in the AC field of that TDM frame. Since a second MPDU (MPDU2) is ready for transport, the ISTE will assert R=1 in that same frame period that SOF was set to "1." If MPDU1 is continued to be transmitted in subsequent TDM frames, the ISTE will continue to also set SOF=0 and R=1 for those TDM frames carrying the MPDU1 payload.

c) Upon receiving SOF=1 for the frame carrying the first fragment of MPDU1, and the R=1 requesting delivery for MPDU2, the AU shall set the grant bit to "1" indicating that a resource is available to also receive MPDU2. The AU shall continue to send G=1 in subsequent TDM frames as long as fragments of MPDU1 arrive with SOF=0.

d) The ISTE will finish sending the last fragment of the MPDU1, and will commence to send the first fragment of the second MPDU (MPDU2). This first fragment of the MPDU2 shall have the accompanying SOF bit set to "1" to indicate the start of a new MPDU. Since there are no more MPDUs queued for transmission, the ISTE will assert R=0 as it delivers payload fragments of MPDU2. The SOF bit will be set to "0" in these subsequent deliveries of the payload fragments for MPDU2.

e) Upon receiving the last payload fragment of MPDU1 and observing that the SOF=1 has been set to indicate the start of another frame transmission, the AU will transmit G=1 in the next TDM frame period to indicate that it has another resource available (potentially for MPDU3).

f) The ISTE will finish sending MPDU2, and since no further MPDUs are pending transmission, the ISTE shall send R=0 and SOF=0 in subsequent "idle" TDM payload frames.

g) As the AU completes the reception of the second MPDU, since it has another receive resource available, it asserts G=1 in the next TDM frame sent back toward the ISTE.

8.3.1.2.7 Priority requests

The prioritization of requests in each ISTE facilitates users to accommodate multiple media service priorities. Functionally, user service prioritization is outside the scope of this standard. However, architecturally the IEEE 802.9 interface will support notification of the priority of the ISTE requests toward the AU. This is particularly useful for the architecture wherein multiple ISTEs are connected to a single AU, and there is contention for the AU receive buffer resources. The AU architecture requires resolving contention based upon the individual request priority.

For single attached AUs (i.e., only one ISTE is connected to the AU), the request priority has no significance within the IEEE 802.9 environment. The timing diagram for the request/grant protocol remains the same even for the prioritized requests, since the IEEE 802.9 AU will receive only one request from each ISTE at any given time. The position of the request priority bits is illustrated in figure 6-7.

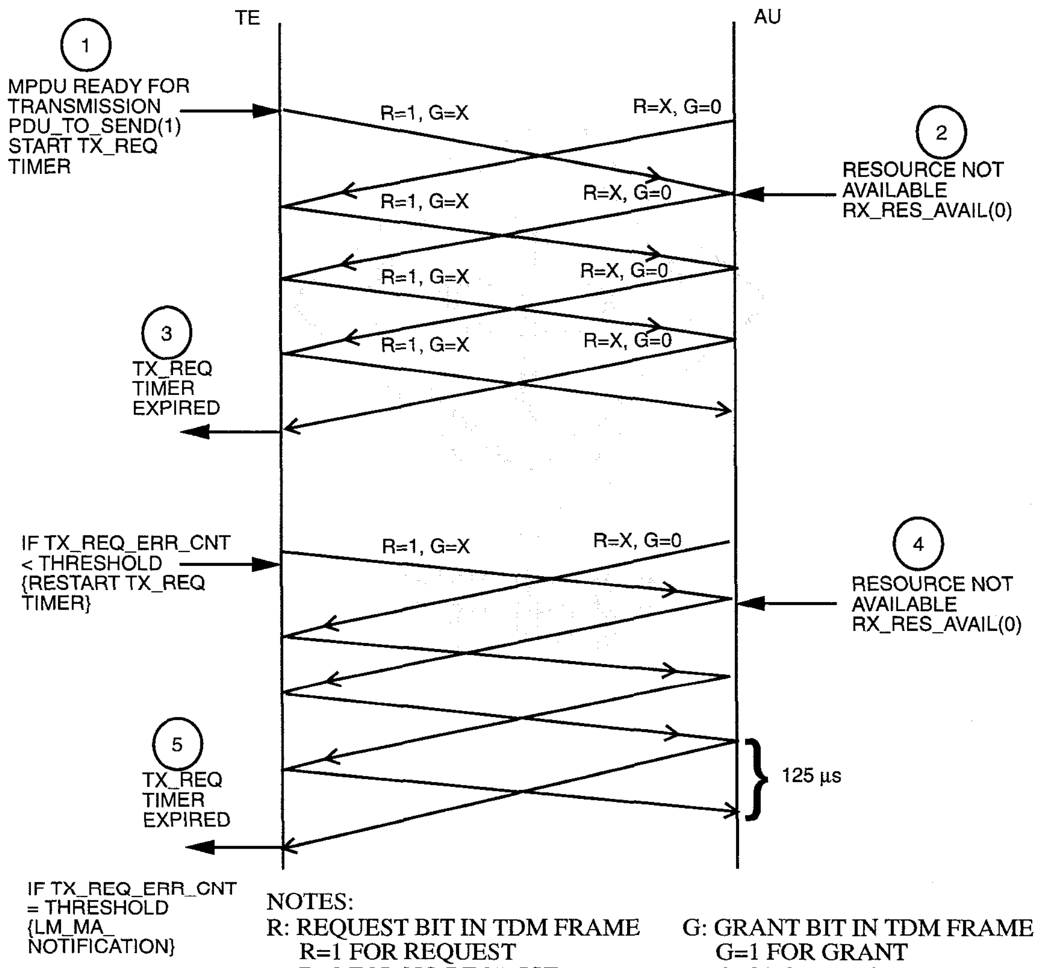
8.3.1.2.8 No grant issued case

There may be the situation where the AU does not have an available receive resource. In this case, the AU will respond with the grant bit set to "0." If the AU does not respond with a grant bit set to "1" over several TDM frame periods, eventually the request transmitter FSM's TX_REQ timer will expire. Upon expiration of the TX_REQ timer, the FSM will increment the TX_REQ_ERR counter. Next, the FSM will check to see if the value of this counter has exceeded a threshold.

a) If the TX_REQ_ERR counter has not exceeded the threshold, the FSM will issue another request and it will restart the TX_REQ timer.

b) If the TX_REQ_ERR counter has now exceeded the threshold, the FSM will issue an LM_MA_NOTIFICATION to the management entity to alert it of the situation.

Figure 8-7 illustrates the timing diagram for the situation when the AU does not issue a grant.



NOTES:
R: REQUEST BIT IN TDM FRAME
R=1 FOR REQUEST
R=0 FOR NO REQUEST
R=X FOR DON'T CARE

G: GRANT BIT IN TDM FRAME
G=1 FOR GRANT
G=0 FOR NO GRANT
G=X FOR DON'T CARE

Figure 8-7—No grant is issued for a request timing diagram

8.3.1.2.9 Grant receive in error case

Whenever the AU sends a grant bit set to "1" (G=1) in a TDM frame, the ISTE can send an MPDU in the next frame period. However, due to channel errors, if the ISTE receives G=0 (even though the AU sent G=1), this will cause MPDU transmission to be delayed by one TDM frame period. The ISTE will have to wait until the next incoming TDM frame period before it observes the G=1.

If the AU has sent G=0 indicating that there is no resource available for receiving an MPDU, but the ISTE actually received a TDM frame with G=1 (due to an error in transmission), and begins transmitting an MPDU payload fragment, this MPDU is likely destined to be ignored by the AU. Retransmission of the lost frame will be handled by the error detection and retransmission procedures.

Handling of the "grant received in error" situation is part of the functionality of the MAC sublayer protocol and is not necessarily a layer management action. However, repeated grant failures will be handled as part of the management actions.

The tx-req-timer is used to keep track of how long R=1 is being sent without receiving the G=1 from the AU. When this timer expires, the tx-req-errcnt is incremented. If the tx-req-errcnt timer expires, then a management action is required.

8.3.1.2.10 Start of frame (SOF) received in error

When the AU has issued a G=1, the ISTE will start sending the MPDU in the next available TDM frame. It will simultaneously issue the SOF=1. If, due to link facility errors, the SOF bit is received as a "0," the new MPDU, which has begun transmission, is lost. Similarly, if the SOF=0 signal is sent during transmission of subsequent payload delivery of the MPDU, but the SOF is received as a logic "1," then the MPDU will be received incorrectly. Retransmission of the lost MPDU will be handled by the error detection and retransmission procedures.

Handling of the "SOF received in error" situation is part of the MAC sublayer protocol and is not a layer management action. However, repeated reception of SOF errors will be handled as part of the management actions.

The rx-grant-timer keeps track of how long the grant is asserted without receiving the SOF. When the timer expires, then a management action is required.

8.3.1.2.11 Resource allocation resolution for multiple ISTEs connected to a single AU

When shared resources are allocated to multiple ISTEs based on requests for MPDU transmission, the AU has to resolve contention if a common resource needs to be allocated to MPDU transmissions from multiple ISTEs. The contention resolution algorithm employed by the AU may be based on enabling higher priority requests to be processed prior to lower priority requests. If more than one request is of the same priority, the AU may employ a local contention mechanism that is dependent on the AU implementation.

8.3.1.2.12 Request/grant protocol using multiple AC fields in a TDM frame

When higher rate physical channels between the ISTE and the AU are used, it is possible to negotiate multiple AC fields to be distributed across the TDM frame. The detailed procedures for negotiation and the number of AC fields in a single TDM frame are a function of the HMUX sublayer. Figure 8-21 illustrates a representative example of the timing diagram and the request/grant protocol operation when multiple MPDUs are transmitted in a single TDM frame. This figure assumes that the TDM frame transmissions between the AU and the ISTE are such that the AC fields are synchronized in time.

The following steps are useful in describing the request/grant procedure using multiple AC fields:

a) The first AC field of the TDM frame from the AU to ISTE has G=1 indicating that the AU has resources available to receive one MPDU.

b) The ISTE sets SOF=1 to start transmitting MPDU1 following the AC field. It will also set R=1 to indicate that MPDU2 is ready for transmission.

c) The AU will continue to place G=1 in the second AC field for MPDU1 since it has not received SOF=1 yet.

d) The ISTE sets SOF=0 and R=1 in the third AC field since the MPDU1 transmission will further continue following this AC field.

e) The AU sets G=1 in the third AC field indicating that it has resources available to receive MPDU2. Note that the transmission of MPDU1 from the ISTE to the AU has completed before the fourth AC field in the TDM transmission from the ISTE to the AU.

f) The ISTE sets SOF=1 in the fourth AC field to begin transmitting MPDU2 since G=1 was received in the third AC field of the TDM frame from the AU. It also sets R=1, since the MPDU3 is ready for transmission.

g) The AU sets G=1 in the fourth AC field in the TDM transmission to the ISTE.

h) Note that the MPDU2 transmission continues even after the fifth AC field of the TDM frame from the ISTE. However, since MPDU3 transmission is about to begin in the payload field that follows the fifth AC field, the ISTE sets SOF=1. Also, since there are no pending MPDUs awaiting transmission, the ISTE will set R=0 in the AC field.

The issue of transmission delay between the TDM frame transmissions between the ISTE and the AU and between the AU and the ISTE is discussed in 8.4.

8.3.1.3 Transmitter/receiver model for request/grant FSM

For purposes of simplification, the FSM for the request/grant protocol procedures has been modeled with the transmitter functions separate from the receiver functions. The transmitter FSM and the receiver FSM (within a peer entity) are interconnected by the internal signals. In practice, the receiver and transmitter circuits within a given peer entity (either AU or ISTE) will be tightly interconnected. Figure 8-8 illustrates this relationship. Note that the transmitter functional module actually has responsibility for sending the stream consisting of request and grant bits. However, only the request bit is formed by the transmitter control functional module. The grant bit is formed by the receiver control functional module within a given peer entity.

8.3.2 FSM for sending request signals

Figure 8-9 illustrates the state transition diagram. Tables 8-1a and 8-1b describe the state transitions for the FSM which controls the transmission of the request signals (REQUEST BIT) in the AC field, and initiates the transmission of an MPDU. The basic operation of the state machine can be modeled as an FSM consisting of three states:

a) INACTIVE
b) AWAIT_MPDU
c) REQUEST

The following three subclauses contain a description of the actions, events, and transitions of the FSMs.

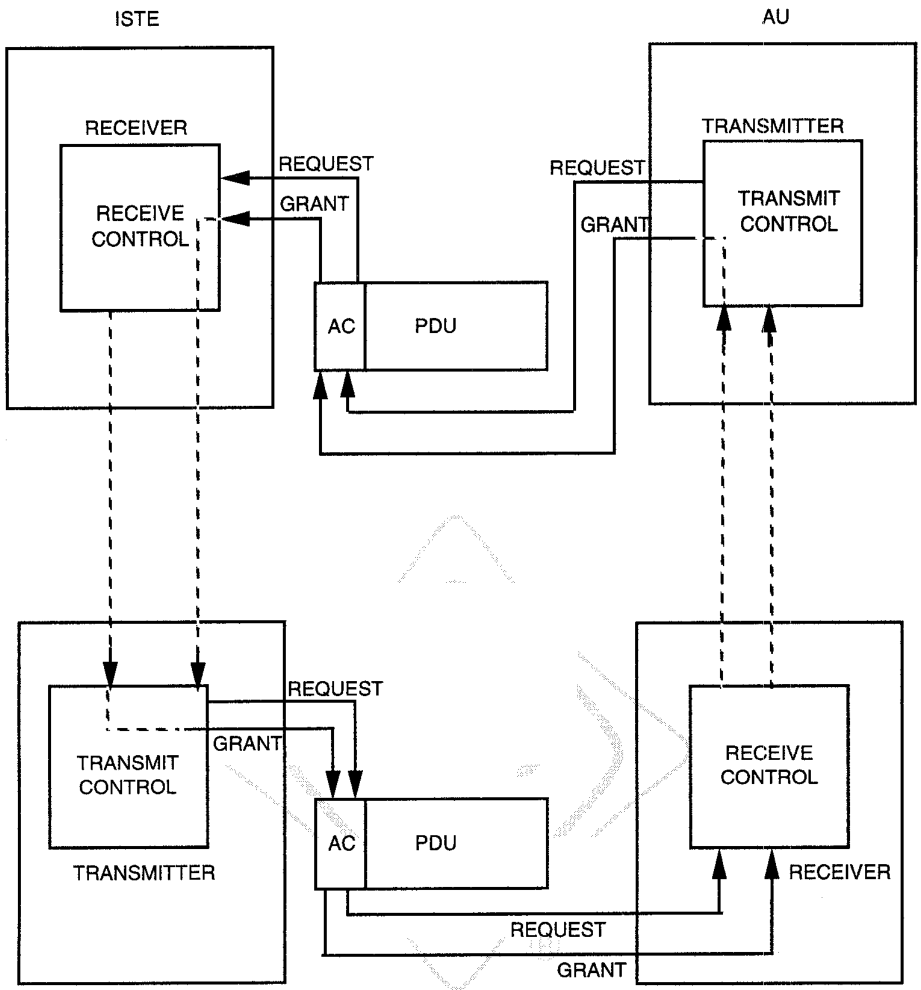


Figure 8-8—Transmitter/receiver model for request/grant FSM

8.3.2.1 INACTIVE state operation

The INACTIVE state is the deactivated condition. In this state, the transmitter FSM will not respond to any stimulus except for messages from its LME. While in this state, there is no data exchange permitted from the ISTE to the AU.

Upon receipt of the management signal, LM-MA-ACTION(activate), the FSM is transitioned to the AWAIT_MPDU state. The LM-MA-ACTION signal contains a complete set of information for the setting of variables and timers associated with this FSM.

8.3.2.2 AWAIT_MPDU state operation

In this state, the FSM continues to scan the PDU_TO_SEND signal. This signal is an indication of whether there are MPDUs queued for transmission. This signal is set to "0" when the transmit queue is empty. When an MPDU is queued for transmission, the PDU_TO_SEND is set to "1."

Once the FSM is in the AWAIT_MPDU state, it will remain in this state as long as the PDU_TO_SEND variable is "0." When the PDU_TO_SEND is set to "1," the FSM will vector to the REQUEST state.

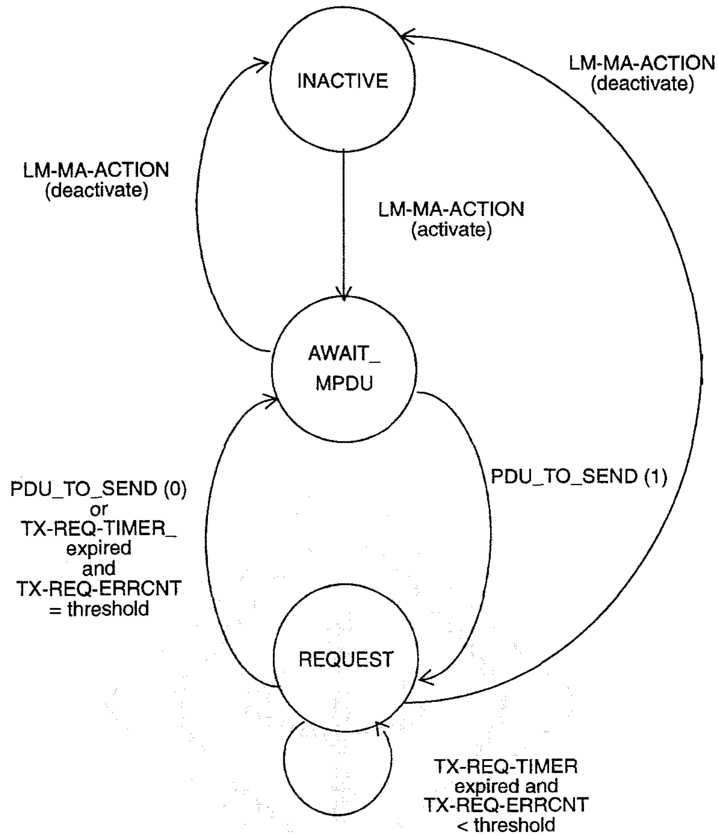


Figure 8-9—FSM state transition diagram for the internal request control

In the AU, when the management status flag IGNORE_GRANT = "1," there is no need to send a request signal in response to PDU_TO_SEND(1). Instead, the FSM shall remain in the AWAIT_MPDU state and it will issue a BEGIN_TX(PDU) signal toward the MPDU transmit control.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

8.3.2.3 REQUEST state operation

The REQUEST state is utilized only for ISTE implementations. This state is not required for AU implementations.

The REQUEST state is the state in which the FSM sends an R=1 to the peer entity in accordance with the PDU_TO_SEND signal from the upper layer. If GRANT_RCVD is "0," the FSM sends a REQUEST bit set to "1" (R=1) to the peer entity. However, if the GRANT_RCVD signal is set to "1," the first queued MPDU will be transmitted immediately. At this point in preparing the first fragment of the MPDU for transmission, if the request FSM sees that there is more than one MPDU queued for transmission, the FSM will set the REQUEST bit to "1" in the TDM frame. The request transmitter will stay in this state until there are no more

MPDUs queued for transmission or until the management error counter, TX_REQ_ERRCNT, reaches the threshold value. Upon the event of first setting R=1, the FSM will start the TX_REQ timer.

Table 8-1a—FSM state transition table for the internal request control for INACTIVE and AWAIT_MPDU states

Current state	Event	Action(s)	Next state
INACTIVE	LM-MA-ACTION (activate)	SET_VARIABLES_AND_ TIMERS	AWAIT_MPDU
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
	PDU_TO_SEND(x)	NO_ACTION	INACTIVE
	GRANT_RECVD(x)	NO_ACTION	INACTIVE
AWAIT_MPDU	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	AWAIT_MPDU
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (RESET)	AWAIT_MPDU
	PDU_TO_SEND(0)	NO_ACTION	AWAIT_MPDU
	PDU_TO_SEND(1) and IGNORE_GRANT=0	SEND_A_REQUEST RESTART_TX-REQ-TIMER	REQUEST
	PDU_TO_SEND(1) and IGNORE_GRANT=1	TX(PDU) (Increment Transmit queue)	AWAIT_MPDU
	GRANT_RECVD(x)	NO_ACTION	AWAIT_MPDU

Table 8-1b—FSM state transition table for the internal request control for REQUEST state

Current state	Event	Action(s)[a]	Next state
REQUEST	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	REQUEST
	LM-MA-ACTION (deactivate)	STOP_TX-REQ-TIMER	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	REQUEST
	PDU_TO_SEND(0)	STOP_TX-REQ-TIMER	AWAIT_MPDU
	PDU_TO_SEND(1)	NO_ACTION	REQUEST
	GRANT_RECVD(0)	SEND_A_REQUEST RESTART_TX-REQ-TIMER	REQUEST
	GRANT_RECVD(1)	STOP_TX-REQ-TIMER TX(PDU) (Increment Transmit Queue)	AWAIT_MPDU
	TX-REQ-TIMER_EXPIRED and TX-REQ-ERRCNT<THRESOLD	TX-REQ-ERRCNT:=TX-REQ-ERRCNT+1 RESTART_TX-REQ-TIMER	REQUEST
	TX-REQ-TIMER_EXPIRED and TX-REQ-ERRCNT=THRESOLD	LM-MA-NOTIFICATION (TX-REQ-ERR-CNT)	AWAIT_MPDU

[a] In the REQUEST state, the AC will continue to set REQUEST_BIT=1 in each successive AC subfield until receipt of the signal GRANT_RECVD(1).

If the GRANT=1 signal is not received before the TX_REQ timer expires, the FSM shall restart the timer TX_REQ, increment the management error counter TX_REQ_ERRCNT, and stay in the REQUEST state. While in the REQUEST state, if the TX_REQ_ERR_CNT counter reaches its threshold, the FSM will send a signal to management, LM-MA-Notification(TX_REQ_ERRCNT), clear the TX_REQUEST flag, and return to the AWAIT_MPDU state.

While in the REQUEST state, if the condition PDU_TO_SEND turns from "1" to "0" before the grant is received, the FSM shall stop timer TX_REQ, and return to the AWAIT_MPDU state. The request transmitter FSM will vector back to the AWAIT_MPDU state. An example of this event would be when layer management decides to stop transmission and rebuild the queue.

As long as the GRANT signal is not received, the variable GRANT_RCVD will remain set to "0." While in this state, if the GRANT bit = 1 is received in the TDM frame, the signal GRANT_RCVD is set to "1."

GRANT_RCVD will be reset to logic "0" only after a TDM frame is received which contains a grant bit equal to "0."

The GRANT_RCVD signal may be involved in the following situations:

a) If GRANT_RCVD=0 and PDU_TO_SEND=1, it indicates that there is an outstanding MPDU awaiting a grant bit = "1."
b) If GRANT_RCVD=1 and PDU_TO_SEND=1, it indicates that an MPDU is being transported. The transmit queue may contain more than one MPDU.
c) If PDU_TO_SEND=0, it indicates that there is no pending MPDU awaiting transmission.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

Note that in figures 8-5 and 8-6 (the multiple request case), the signal PDU_TO_SEND will continue to be set to 1 and TX_REQUEST=1 will remain. After the first MPDU transmission is completed, the TX- REQ-TIMER is restarted, and the variable TX-REQ-ERR-CNT is reset. In addition, the REQUEST bit in the TDM frame is set to "1" for requesting the ability to transmit another MPDU. The peer entity may issue GRANT= 1 in every TDM frame, independent of the actions of this FSM. The identification of the GRANT signal for different MPDUs is in accordance with the procedures described in 8.3.1.2.6.

8.3.3 FSM for sending grant signals at the ISTE

Figure 8-10 and tables 8-2a and 8-2b describe the FSM which controls the transmission of the grant signals at the ISTE. The basic operation of the state machine consists of three states:

a) INACTIVE
b) AWAIT_RES
c) GRANT

Upon receipt of the management signal, LM-MA-ACTION(activate), the FSM vectors to the AWAIT_RES state. The LM-MA-ACTION signal contains a complete set of information for the setting of variables and timers associated with this FSM.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

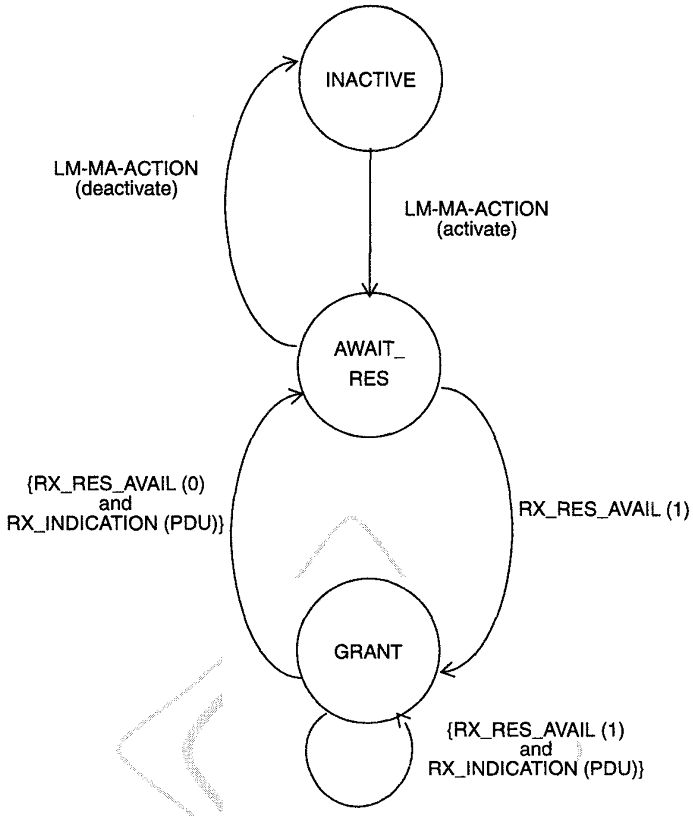


Figure 8-10—FSM state transition diagram for the internal grant control

Table 8-2a—FSM state transition table for the internal grant control for INACTIVE and AWAIT_RES states

Current state	Event	Action(s)	Next state
INACTIVE			AWAIT_RES
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
	RX_RES_AVAIL(x)	NO_ACTION	INACTIVE
AWAIT_RES	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	AWAIT_RES
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (RESET)	AWAIT_RES
	RX_RES_AVAIL(0)	NO_ACTION	AWAIT_RES
	RX_RES_AVAIL(1)	SEND_A_GRANT	GRANT
	RX-INDICATION(PDU)	INCREMENT_RX-RES-ERRCNT	AWAIT_RES

Table 8-2b—FSM state transition table for the internal grant control for GRANT state

Current state	Event	Action(s)	Next state
GRANT	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	LM-MA-ACTION (deactivate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	RX_RES_AVAIL(0) and RX-INDICATION(PDU)	INCREMENT_RX-RES-ERRCNT	AWAIT_RES
	RX_RES_AVAIL(1) and RX-INDICATION(PDU)	DECREMENT_RCV_QUEUE	GRANT

Each ISTE will be initialized such that RX_RES_AVAIL is set to "1." Thus the state machine stays in the GRANT state all the time. This initialization allows the AU to transmit an MPDU to the ISTE whenever it has an MPDU to transmit to the ISTE.

As long as a resource is available to receive an MPDU, i.e., RX_RES_AVAIL = 1, the FSM moves to and stays in the GRANT state.

When resource becomes not available to receive an MPDU, i.e., RX_RES_AVAIL = 0, the FSM moves to and stays in the AWAIT_RES state.

Note that the relationship of the TX_GRANT flag and the GRANT bit of the AC field is as follows:

— Whenever TX_GRANT = "0," the GRANT bit in the AC field is set to "0"
— Whenever TX_GRANT = "1," the GRANT bit in the AC field is set to "1"

While in the GRANT state, upon detection of a start of an incoming MPDU, the status indicator RX_INDICATION(PDU) will be set to "1." At this time there will be one of two actions that the FSM will follow:

— If there is a receive resource available (RX_RES_AVAIL=1), the receive byte counter (RX_BYTE_CNT) will be cleared and the receive queue will be decremented by one. The FSM will remain in the GRANT state.

— If there is no receive resource available (RX_RES_AVAIL=0), the receive resource error counter (RX_RES_ERRCNT) will be incremented by one. The FSM will return to the AWAIT_RES state.

Note that layer management cannot direct that the FSM leave the GRANT state. This is because once a GRANT="1" is issued, the FSM must await the allowed time for the peer-to-peer response to be completed.

8.3.4 FSM for sending grant signals at the AU

This subclause describes examples of the FSMs for the request/grant protocols at the AU. The exact FSM for the AU is dependent on the model assumed for the AU. This subclause describes FSMs for the following two AU models:

a) *Shared buffer model.* Grants signals used for resource allocation.
b) *Dedicated buffer model.* Grants signals used for flow control.

8.3.4.1 Arbitration at the AU—Shared buffer model

This AU model assumes that the resources at the AU are shared by all ISTEs attached to it.

Figure 8-11 illustrates the AU which is based on the shared buffer. The AU's access controller (ACC) and the Buffer Manager manage the allocation and deallocation of buffers to particular ISTEs. The basic request/grant protocol operations in this model are as follows:

a) Each ISTE, which has one or more MPDUs to transmit, sends the request signals to the ACC.

b) The ACC arbitrates the requests from the ISTEs and sends the grant signal to the ISTE to which the associated Buffer Manager allocated some resources for PDU reception. It is possible that more than one ISTE could be transmitting an MPDU to the AU at the same moment in time.

c) The ISTE which has received the GRANT_BIT=1 signal may transmit one complete MPDU toward the AU starting in the P channel. The relationship between MPDUs, the GRANT bit, and the SOF is in accordance with the description in 8.3.1.2.6.

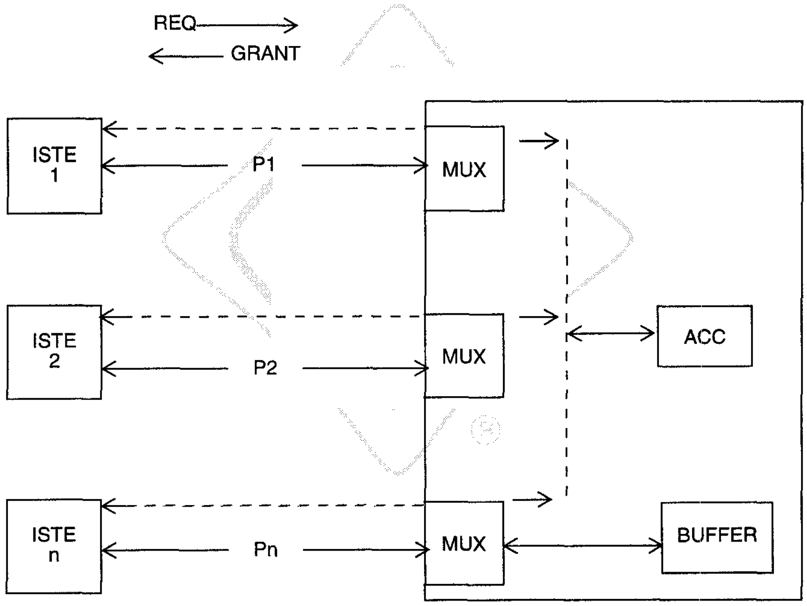


Figure 8-11—Shared buffer model

8.3.4.1.1 FSM for sending grant signals at shared buffer (AU)

Figure 8-12 and tables 8-3a and 8-3b describe the FSM which controls the transmission of the grant signals at the AU. The basic operation of the state machine consists of three states:

a) INACTIVE
b) AWAIT_RES
c) GRANT

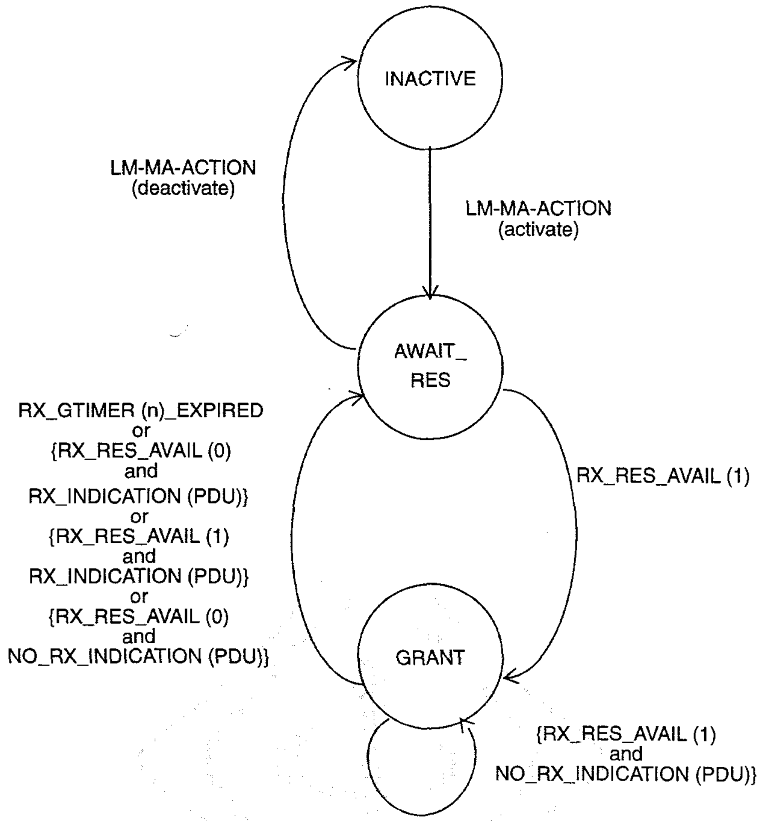


Figure 8-12—FSM state diagram for sending grant—Shared buffer model

For the shared buffer model of the AU, it is important to note that the FSM described in figure 8-12 and tables 8-3a and 8-3b will exist for every active ISTE attached to the AU (i.e., there will exist one state machine instance for each ISTE port).

Upon receipt of the management signal, LM-MA-ACTION(activate), the FSM is transitioned to the AWAIT_RES state. The LM-MA-ACTION signal contains a complete set of information for the setting of variables and timers associated with this FSM.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

With the shared resource model, the AU may or may not have receive resources available. When a resource becomes available (RX_RES_AVAIL=1), the AU's FSM will set the status indicator, TX_GRANT, to "1" for that ISTE instance and vector to the GRANT state. Since the AU does not want to keep these resources allocated forever, it will associate a timer with the GRANT token signal that is issued. When this timer expires, the AU will revoke the GRANT token and stop the timer record. At the end of the timer period, if the ISTE has not sent a packet that would require a receive buffer, then the AU is free to put that resource back into its pool of common receive resources.

As long as a resource is available to receive an MPDU, i.e., RX_RES_AVAIL = "1," the FSM moves to and stays in the GRANT state. As mentioned above, the FSM will start a timer, RX_GTIMER(n), for each grant that is issued.

If a RX_GTIMER(n) expires, the AU will expect that the resource was unused. At this point, the unused GRANT token will be put back into the free receive resource pool. The error counter, RX_GTIMER_ERRCNT, is incremented by one at this point.

Table 8-3a—FSM state transition table for sending grant—Shared buffer model for INACTIVE and AWAIT_RES states

Current state	Event	Action(s)	Next state
INACTIVE	LM-MA-ACTION (activate)	SET_VARIABLES_AND_ TIMERS	AWAIT_RES
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
	RX_RES_AVAIL(x)	NO_ACTION	INACTIVE
AWAIT_RES	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	AWAIT_RES
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (RESET)	AWAIT_RES
	RX_RES_AVAIL(0)	NO_ACTION	AWAIT_RES
	RX_RES_AVAIL(1)	SEND_A_GRANT RESTART_TIMER_RX_ GTIMER(n)	GRANT
	RX-INDICATION(PDU)	INCREMENT_RX-RES_ ERRCNT	AWAIT_RES

Table 8-3b—FSM state transition table for sending grant—Shared buffer model for GRANT state

Current state	Event	Action(s)	Next state
GRANT	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	LM-MA-ACTION (deactivate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	GRANT
	RX_RES_AVAIL(0) and NO_RX_INDICATION(PDU))	STOP_RX_GTIMER(n)	AWAIT_RES
	RX_RES_AVAIL(0) and RX_INDICATION(PDU))	INCREMENT_RX_RES_ ERRCNT STOP_RX_GTIMER(n)	AWAIT_RES
	RX_RES_AVAIL(1) and NO_RX_INDICATION(PDU))	NO_ACTION	GRANT
	RX_RES_AVAIL(1) and RX_INDICATION(PDU))	DECCREMENT_RCV_QUEUE STOP_RX_GTIMER(n)	AWAIT_RES
	RX_GTIMER(n)_EXPIRED	INCREMENT_RX_GTIMER_ ERRCNT	AWAIT_RES

The relationship of the TX_GRANT status flag and the grant bit of the AC field is as follows:

— Whenever TX_GRANT = "0," the grant bit in the AC field is set to "0" (for that ISTE instance)
— Whenever TX_GRANT = "1," the grant bit in the AC field is set to "1" (for that ISTE instance)

8.3.4.2 Flow control at the AU—Dedicated buffer model

This AU model assumes that for each ISTE attached to it, there is a dedicated buffer allocated with additional functions such as the address filtering (ADF). This model is illustrated in figure 8-13. Since buffers are allocated to each ISTE, it is not necessary for the AU to wait for a request from the ISTE before issuing a GRANT, thus minimizing the packet latency.

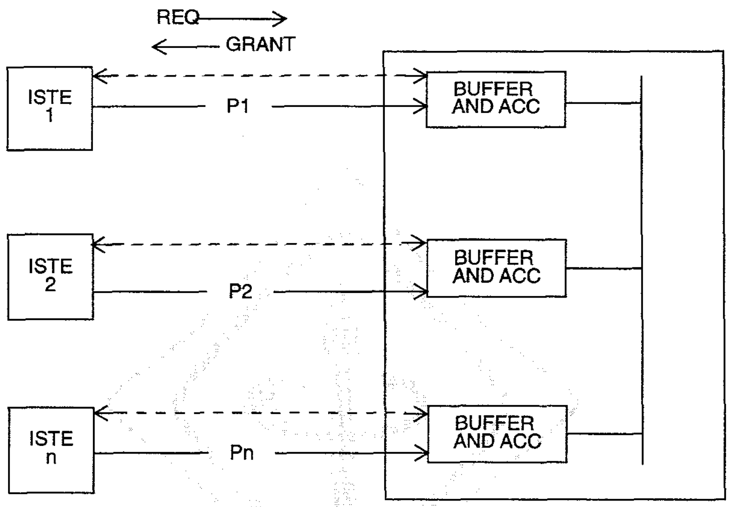


Figure 8-13—AU—Dedicated buffer model

The basic operations are as follows:

a) The AU sets TX_GRANT = "1" if there is sufficient buffer space available
b) The AU sets TX_GRANT = "0" if there is not sufficient buffer space available

In this case the FSM is identical with that of the ISTE GRANT generation. Figure 8-10 shows the FSM state diagram. For a complete description of operation, refer to 8.2.3.

8.3.5 Management initialization procedures for IGNORE_GRANT

The initialization of IGNORE_GRANT differs between the ISTE and the AU. This was done to allow the greatest range of cost/performance designs for AUs, as well as to minimize the impact on the AU of not getting GRANT from an ISTE during periods of high traffic.

8.3.5.1 AU initialization procedure for IGNORE_GRANT

By default, the AU management entity directs the AU to initialize to a state where it will ignore GRANT. This means the AU can send packets to the ISTE without waiting for a GRANT.

8.3.5.2 ISTE initialization procedure for IGNORE_GRANT

By default, the ISTE management entity directs the ISTE to initialize to a state where it pays attention to the GRANT bits coming from the AU. The AU must send a GRANT to the ISTE before the ISTE can send an MPDU.

8.3.6 FSM for sending MPDUs

Figure 8-14 and tables 8-4a and 8-4b describe the FSM which controls the transmission of an MPDU. The basic operation of the state machine can be modeled as an FSM consisting of three states:

a) INACTIVE
b) IDLE
c) SEND

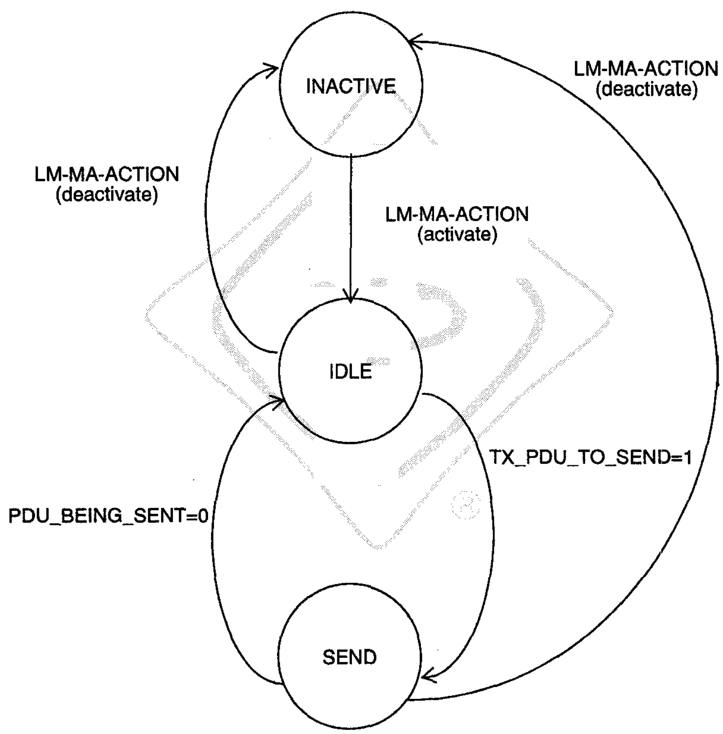


Figure 8-14—FSM state transition diagram for sending MPDUs

Upon receipt of the management signal, LM-MA-ACTION(activate), the MPDU transmit control FSM is transitioned to the IDLE state. The LM-MA-ACTION signal contains a complete set of information for the setting of variables and timers associated with this FSM.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

Once the FSM is in the IDLE state, whenever the MAC access control has an MPDU queued for transmission and has received a GRANT_BIT=1, the FSM sends the message TX(pdu) which serves to increment the transmit queue that is serving the MPDU transmit control module. Whenever there is one or more PDUs queued, this activates the internal signal TX_PDU_TO_SEND to logic value "1." In response, the MPDU transmit control FSM will vector to the state SEND. It stays in this state as long as the PDU_BEING_SENT = “1” condition is valid.

While in the SEND state, when the last bit of the PDU has been sent, the variable PDU_BEING_SENT turns from “1” to “0.” In response to this event, the MPDU transmit control FSM shall stop transmitting, clear the PDU_BEING_SENT flag, decrement the transmit queue, and return to the IDLE state.

While in the SEND state, upon the receipt of the management signal, LM-MA-ACTION(deactivate), the MPDU transmit control FSM is transitioned to the INACTIVE state.

Table 8-4a—FSM state transition table for sending MPDUs for INACTIVE and IDLE states

Current state	Event	Action(s)	Next state
INACTIVE	LM-MA-ACTION (activate)	SET_VARIABLES_AND_ TIMERS	IDLE
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
	TX_PDU_TO_SEND=x	NO_ACTION	INACTIVE
	PDU_BEING_SENT=x	NO_ACTION	INACTIVE
IDLE	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	IDLE
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (RESET)	IDLE
	TX_PDU_TO_SEND=0	NO_ACTION	IDLE
	TX_PDU_TO_SEND=1	START_TRANSMIT_PDU PDU_BEING_SENT:=1	SEND
	PDU_BEING_SENT=x	NO_ACTION	IDLE

Table 8-4b—FSM state transition table for sending MPDUs for SEND state

Current state	Event	Action(s)	Next state
SEND	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	SEND
	LM-MA-ACTION (deactivate)	STOP_TRANSMIT_PDU PDU_BEING_SENT:=0	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	SEND
	TX_PDU_TO_SEND=x	NO_ACTION	SEND
	PDU_BEING_SENT=0	STOP_TRANSMIT_PDU DECREMENT_XMT_QUEUE	IDLE
	PDU_BEING_SENT=1	CONTINUE_TRANSMIT	SEND

8.3.7 FSM for receiving MPDUs

Figure 8-15 and tables 8-5a and 8-5b describe the FSM which controls the reception of an MPDU. The basic operation of the state machine can be modeled as an FSM consisting of three states:

a) INACTIVE
b) IDLE
c) RECEIVE

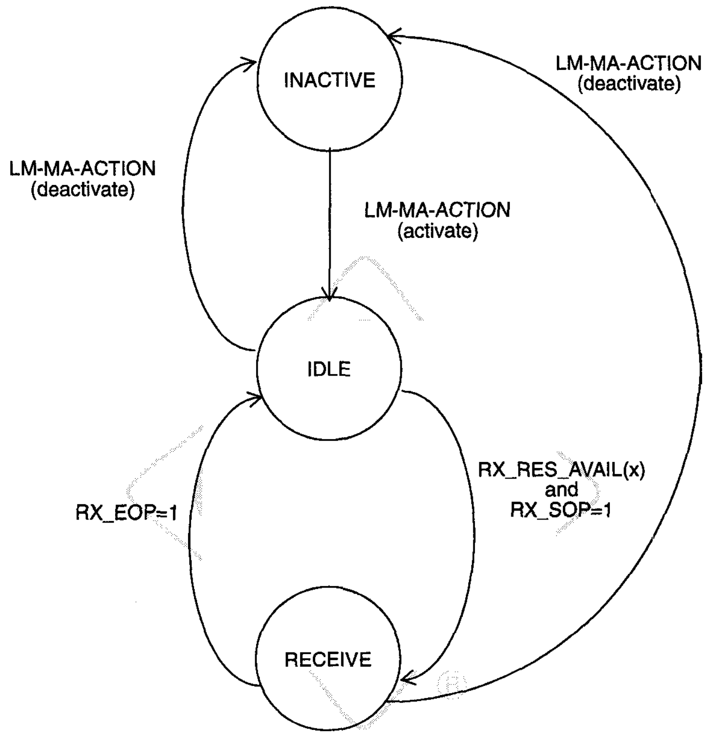


Figure 8-15—FSM state transition diagram for receiving MPDUs

Upon receipt of the management signal, LM-MA-ACTION(activate), the MPDU receive control FSM is transitioned to the IDLE state. The LM-MA-ACTION signal contains a complete set of information for the setting of variables and timers associated with this FSM.

Upon receipt of the management signal, LM-MA-ACTION(deactivate), the FSM is transitioned to the INACTIVE state.

Once the MPDU receive control FSM is in the IDLE state, whenever the MAC access control has an available buffer and has received an RX_SOP=1 indication, the FSM begins the reception process. In response, the MPDU receive control FSM will vector to the RECEIVE state. It stays in this state until the RX_EOP=1 indication is received or a packet length violation is reached.

While in the RECEIVE state, once the reception process has started, the MPDU receive control FSM will scan for the presence of an end of packet (RX_EOP=1) indication. The MPDU receive control FSM will be responsible for counting bytes and storing that count in RX_BYTE_CNT.

Table 8-5a—FSM state transition table for receiving MPDUs for INACTIVE and IDLE states

Current state	Event	Action(s)	Next state
INACTIVE	LM-MA-ACTION (activate)	SET_VARIABLES_AND_ TIMERS	IDLE
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
	RX_RES_AVAIL=x	NO_ACTION	INACTIVE
	RX_SOP=x	NO_ACTION	INACTIVE
	RX_EOP=x	NO_ACTION	INACTIVE
IDLE	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	IDLE
	LM-MA-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (RESET)	IDLE
	RX_SOP=0	NO_ACTION	IDLE
	RX_RES_AVAIL=0 and RX_SOP=1	INCREMENT_RX_RES_ ERRCNT	IDLE
	RX_RES_AVAIL=1 and RX_SOP=1	RX_BYTE_CNT:=0 DECREMENT_RCV_QUEUE	RECEIVE
	RX_EOP=x	NO_ACTION	IDLE

Table 8-5b—FSM state transition table for receiving MPDUs for RECEIVE state

Current state	Event	Action(s)	Next state
RECEIVE	LM-MA-ACTION (activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	RECEIVE
	LM-MA-ACTION (deactivate)	STOP_RECEIVE_PDU DISCARD_PDU_BEING_ RECEIVED RESET-RECEIVE_QUEUE	INACTIVE
	LM-MA-ACTION (reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	RECEIVE
	RX_RES_AVAIL=x	NO_ACTION	RECEIVE
	RX_SOP=x	NO_ACTION	RECEIVE
	RX_EOP=0	CONTINUE_TO_RECEIVE	RECEIVE
	RX_EOP=1	STOP_RECEIVE IF(RX_BYTE_CNT>RX_MAX_ MPDU) THEN_INCREMENT_RX_ MAX_ERRCNT IF(RX_BYTE_CNT>RX_MIN_ MPDU) THEN_INCREMEN-T_RX_MIN_ERRCNT	IDLE

Once the RX_EOP indication is set, the receiver will then check the received byte count. Errors will be counted as follows:

a) If the received byte count (RX_BYTE_CNT) is greater than the maximum MPDU size, the error counter RX_MAX_MPDU will be incremented.
b) If the received byte count (RX_BYTE_CNT) is less than the minimum MPDU size, the error counter RX_MIN_MPDU will be incremented.

At the end of error checking, the MPDU received indication will be raised and the MPDU receive control FSM will vector back to the IDLE state.

While in the RECEIVE state, upon receipt of the management signal, LM-MA-ACTION(deactivate), the MPDU receive control FSM is transitioned to the INACTIVE state.

8.4 Timing relationship of TDM frames exchanged between the ISTE and the AU

If the transmission of the request bits of the AC field in the TDM frame from the ISTE to the AU is accomplished after the reception of the grant bit in the AC field in the TDM frame sent by the AU to the ISTE, there may be specific advantage with respect to the performance of the request/grant protocol. However, this requires that the TDM frame transmission from the ISTE to the AU may be delayed with respect to the TDM frame from the AU to the ISTE. The amount of the delay is a function of the propagation delay between the ISTE and the AU, and is implementation dependent. As such, the exact delay is not specified as part of this standard.

The basic request/grant protocol will function on both implementations whether or not delay is implemented.

It should be noted that delay of ISTE TDM frame transmission to the AU may impact the performance of the AU if IGNORE_GRANT is not implemented in the AU.

Figure 8-16 illustrates the use of multiple AC fields in a single TDM frame.

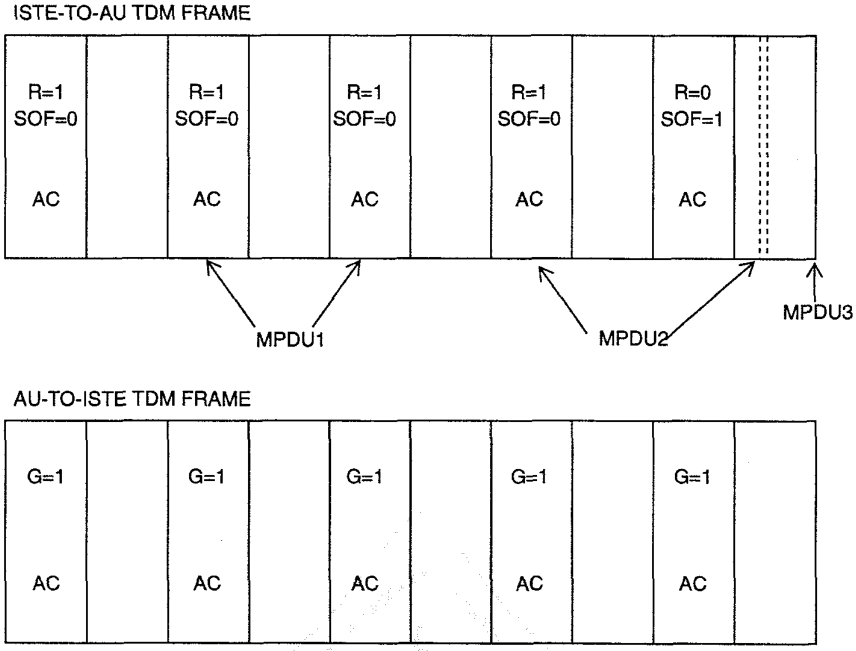


Figure 8-16—Request/grant protocol with multiple AC fields in a TDM frame

9. Detailed specification of the Physical (PHY) Layer

9.1 Overview

The PHY Layer is responsible for establishing, maintaining, and terminating communication between the peer PHY Layer entities. As described in 5.3.2, the PHY Layer is subdivided into hybrid multiplexing (HMUX), physical signalling (PS), and physical medium dependent (PMD) sublayers. An overview of the functions at each sublayer and the signals that cross the sublayer boundaries, to adjacent sublayers and to management, are shown in figure 9-1. In this chapter, detailed specifications related to those sublayer functions and sublayer signals are presented. The IEEE 802.9 interface specifies two particular PMD sublayers:

a) A PMD for the lower rate (4.096 Mb/s) and longer reach (450 m) applications
b) A PMD for the higher rate (20.48 Mb/s) and shorter reach (135 m) applications

Future IEEE 802.9 ISLAN architectures are expected to support other line rates and PMDs that will encompass line rates that are other multiples of 4.096 Mb/s.

This chapter also specifies the connector, the medium cable characteristics, the PHY Layer initialization and maintenance procedures, and the environmental specifications.

9.2 HMUX sublayer

Figure 9-2 shows a detailed model of the transmitter/receiver for the HMUX sublayer in the ISTE. Figure 9-3 shows a detailed model of the transmitter/receiver for the HMUX sublayer in the access unit (AU).

9.2.1 Multiplexing

The multiplexing and demultiplexing shall be performed based on the masked clock signals appropriately generated and timed with respect to the physical time division multiplexer (TDM) frame structure defined in 6.5.2. The service information being multiplexed and demultiplexed are the P, D, B, and C channels. The overhead fields [the P channel access control (AC), hybrid multiplexer control (HMC), and TDM_MTN] and the D and B channels are multiplexed into the leading fixed portion of the TDM frame, whereas the P and C channels are variably assigned within the P/C payload of the TDM frame (see figure 9-2).

The PHY service primitives defined in 7.1.1.5 have associated arguments. One argument, SDU, corresponds to those data bits that are multiplexed and demultiplexed, where the symbol could be either a Binary 0 or a Binary 1. A second argument, Status-Value, designates the validity (or invalidity) of the received data_symbol. The third PhSAP-Id argument provides at the PHY isochronous interface (see 9.2.4) a value that may be used by the PHY service user to multiplex and demultiplex separate user service channels. Note that the HMUX sublayer does not support actual channel multiplexing. It offers through the PHY isochronous interface, received and transmitted channel identifiers a "virtual" channel multiplexing service. The time slot mapping from PhSAP to physical (isochronous timed) channel bandwidth may be either of the following:

a) Prefixed through some initial management configuration agreement
b) Dynamically determined through the exchange of management information

The service primitives also pass isochronous information relative to isochronous clock synchronization.

The external interfaces of the HMUX sublayer are illustrated in figure 9-4.

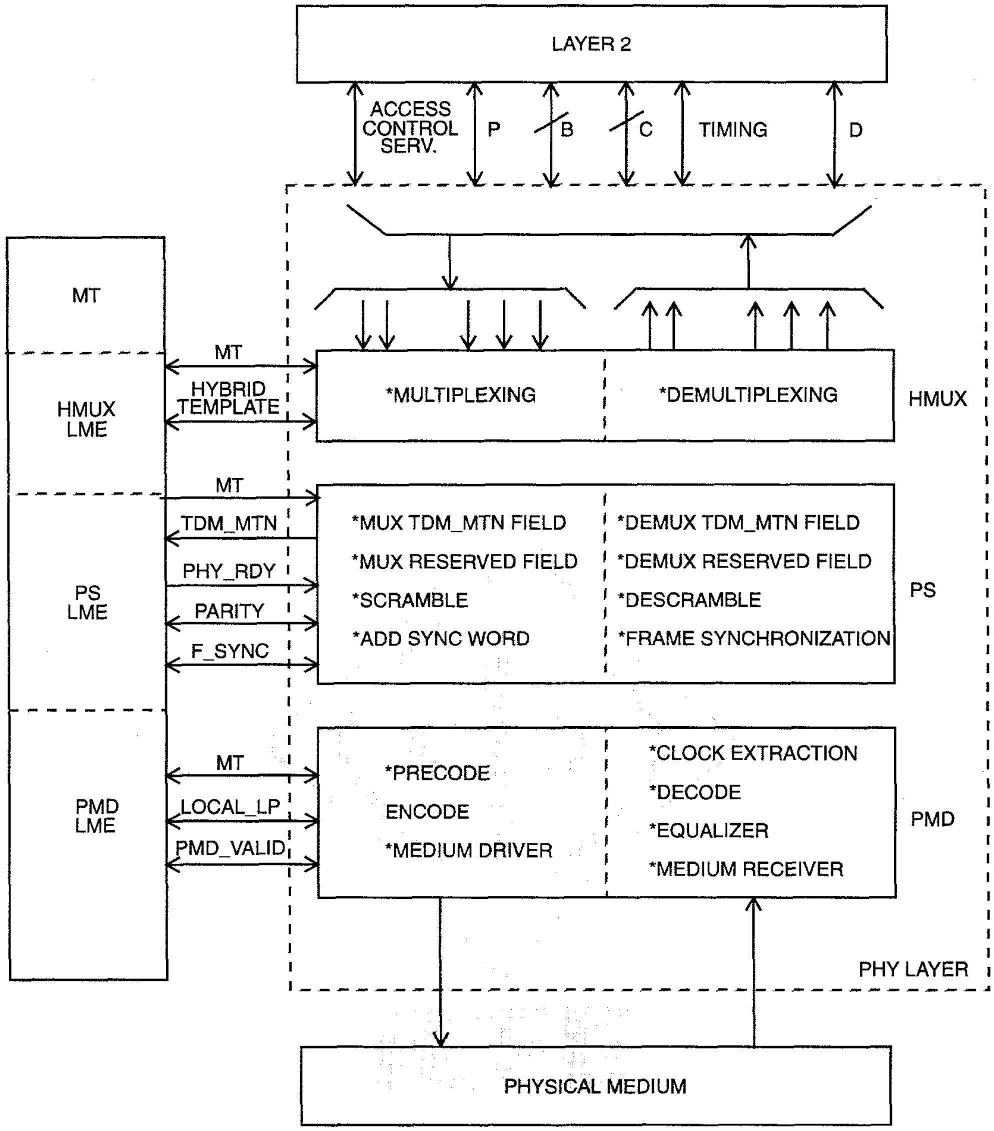


Figure 9-1—General model of the PHY Layer

9.2.2 Hybrid multiplexer control (HMC) field summary

The IEEE 802.9 integrated services LAN (ISLAN) interface supports multiple isochronous applications as well as provides an IEEE 802-based packet service and other framed services over the P channel. The principal negotiation mechanism for providing configuration management of this payload component is through the use of the CCITT Q.93x family of signalling protocols (see Chapter 11). However, it is important to note that the reconfiguration of the bandwidth in the TDM frame must be synchronized between the AU and the ISTE. The HMC field contains the control information that effects the synchronization and determines the D channel rate (16 kb/s or 64 kb/s).

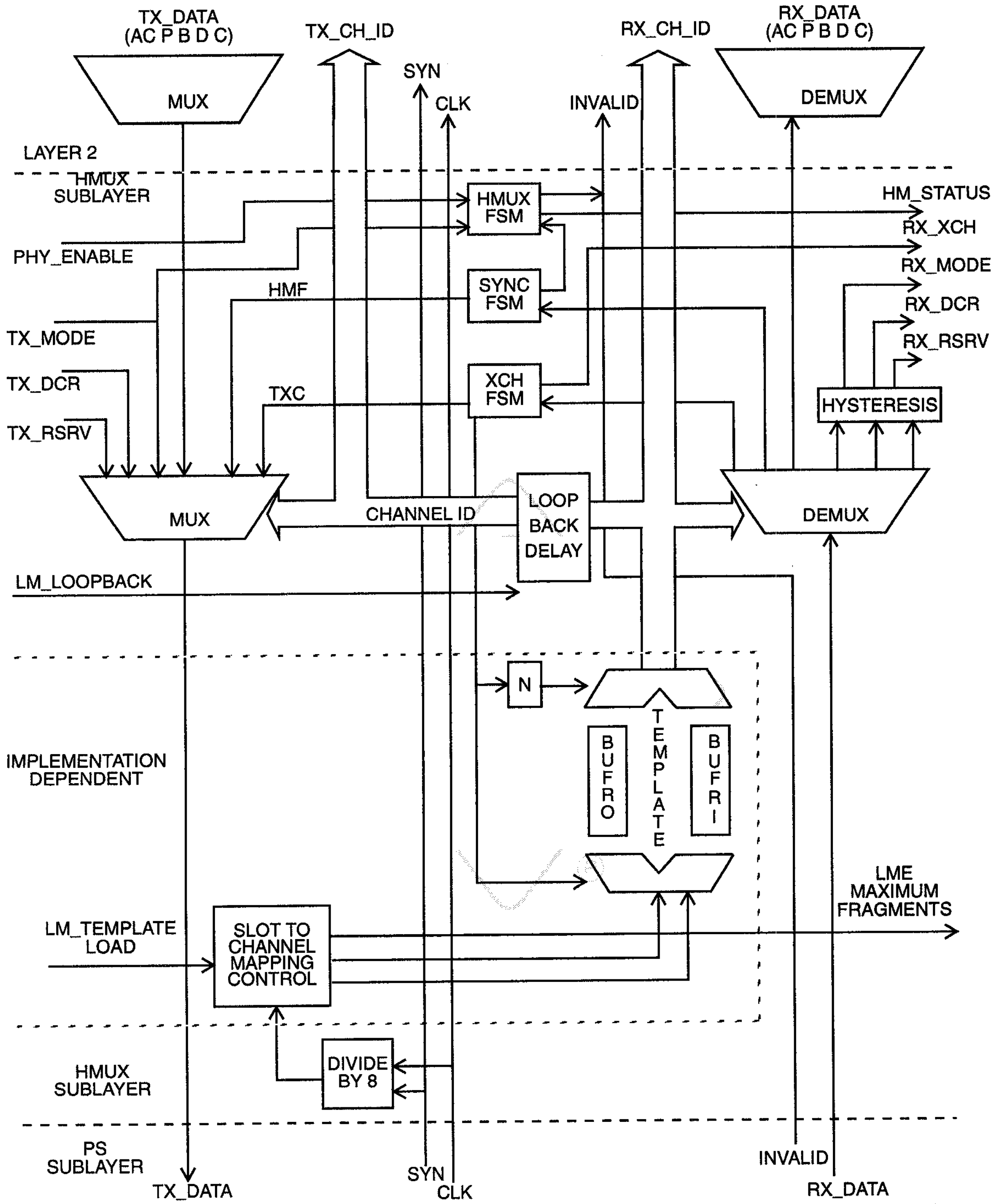


Figure 9-2—Model of the transmitter/receiver for the terminal equipment HMUX sublayer

With the exception of the information that is processed by the PHY Layer, the complete received TDM information, including the SYN and TDM maintenance fields, is passed over the PHY isochronous interface with the appropriate channel ID values.

This HMC field resides in the third octet of the TDM frame. The field is encoded as shown in figure 9-5. The summarized list of functions is given in table 9-1. The functionality of these bits of the HMC field is described in the following subclauses.

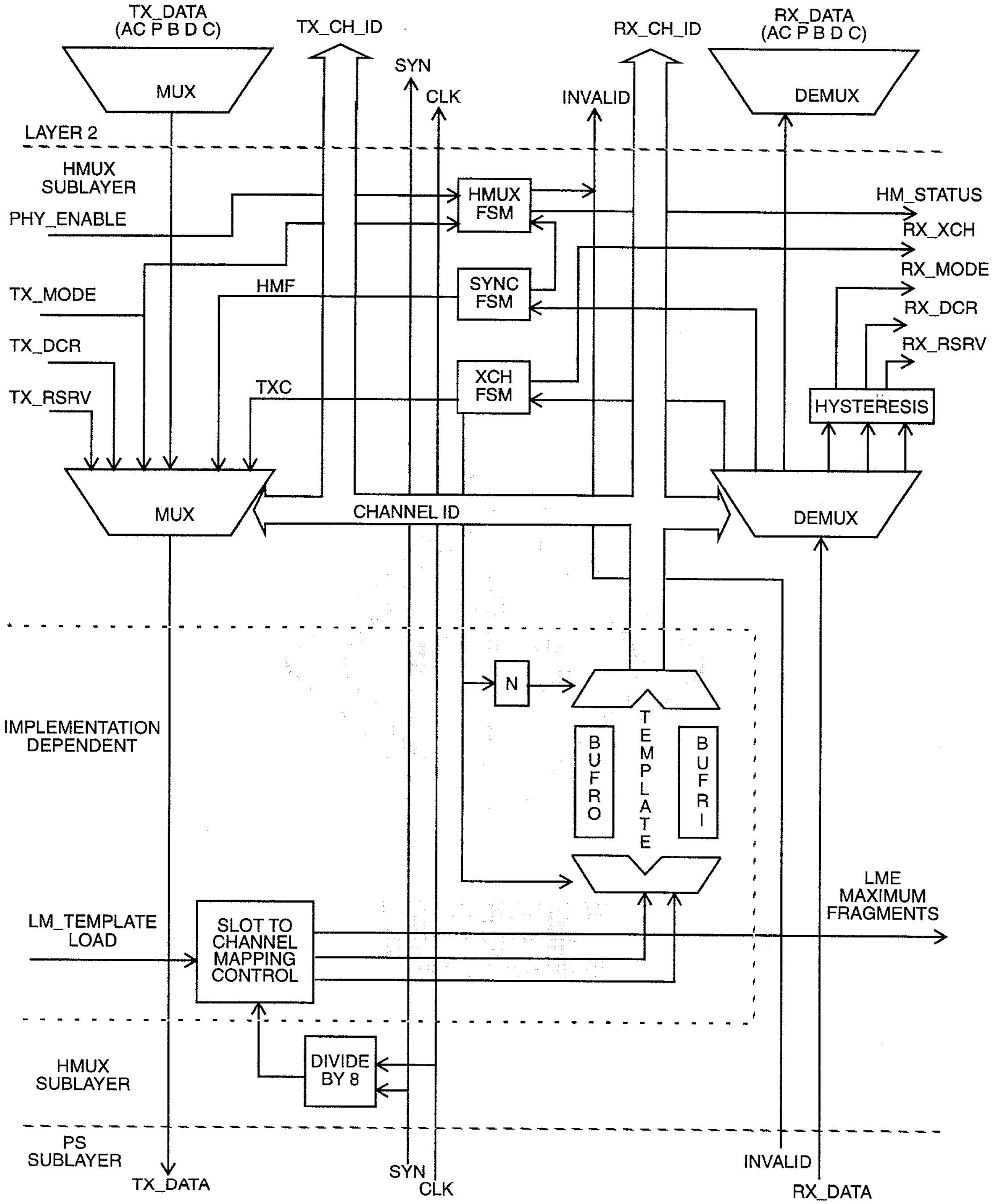


Figure 9-3—Detailed model of the transmitter/receiver for the AU HMUX sublayer

Juniper Ex 1032 - 146
Juniper v Chrimar
IPR2016-01399

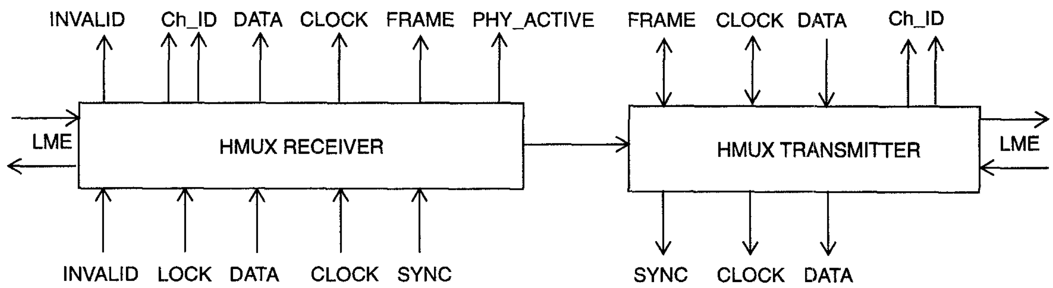


Figure 9-4—HMUX sublayer external interfaces

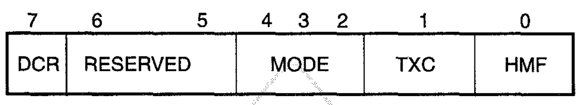


Figure 9-5—TDM frames's HMC field

Table 9-1—Functionality of HMC field

Bit	Name	Function	Value
7	DCR	D channel rate select	$0=D_{16}$ $1=6_{64}$
6, 5	RESERVED	Reserved	Transmitted as 0 and ignored upon receipt
4, 3, 2	MODE	Template identification MODE bits	0 = Mode 0 1 = Mode 1 2 = Mode 2 3 = Mode 3 4 = Reserved 5 = Reserved 6 = Reserved 7 = Reserved
1	TXC	Template exchange notification	0 = Keep templates 1 = Exchange templates
0	HMF	HMUX multiframing	0 = No multiframing bit 1 = Multiframing bit

9.2.2.1 Signal hysteresis to protect the MODE, DCR, and reserved bits

Together the hybrid multiplexer multiframing (HMF) and the template exchange notification (TXC) signals form part of a dynamic bandwidth management process, and their integrity is contained within this process. The other signals within the HMC field (MODE, DCR, RESERVED) are protected from the impact of link errors by a hysteresis mechanism in which there must be three identical signals in consecutively received HMC fields to effect a change of status at the receiver.

9.2.2.2 D channel rate

The D channel rate select (DCR) bit is a single control bit employed in the HMC field to notify the HMUX sublayer as to whether the D channel operates at a 16 kb/s (D_{16}) or a 64 kb/s (D_{64}) rate. When the D_{16} option is selected, the HMUX sublayer shall apply the transmitted information in the two least significant bits of the D channel. The six more significant bits of the D channel shall be set to "0" as this bit stream is applied toward the PMD transmitter. The transmitted DCR bit is sent by the HMUX sublayer under the control of the layer management signal, LM_TX_DCR. The received DCR bit is sent by the HMUX sublayer to layer management via the signal, LM_RX_DCR.

9.2.2.3 RESERVED bits

These bits are reserved for future standardization use by the IEEE 802.9 ISLAN Working Group. Since the entire HMC field is considered to be delivered to the peer entity as part of the service agreement of the HMUX sublayer to its layer management entity (LME) and its adjoining Layer 2, the contents of this field shall be delivered to the peer entity. These bits shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU.

The use of these bits for a proprietary application may lead to incompatibilities with future versions of this standard.

The transmitted values are passed across the LME interface to the HMUX sublayer by the signal, LM_TX_RSRV. Received HMC field bits are passed to layer management by the signal, LM_RX_RSRV.

9.2.2.4 Template identification MODE bits

Template identification MODE bits are used to designate a bandwidth allocation format or "mode" by which the ISTE and AU communicate for the allocation of bearer service across the IEEE 802.9 interface. The modes describe degrees of flexibility with respect to how bandwidth is allocated. These modes vary from a single service (IEEE 802 only) to a more involved, fully dynamic CCITT Q.93x message-based approach for bandwidth management.

The purpose of the modes is to allow lower functionality equipment to enjoy the utilization of the channel structures without having to engage in a fully dynamic network management scheme.

Each mode of operation can be described as a template of the channel configuration of the IEEE 802.9 TDM frame operation. The location of the C channel(s) isochronous bandwidth effectively determines the P channel bandwidth since the unallocated isochronous bandwidth may revert to the P channel MAC transport.

A functional description of the modes is given in annex E.

The use of these reserved bits for a proprietary application may lead to incompatibilities with future versions of this standard.

The transmitted values are passed across the LME interface to the HMUX sublayer by the signal, LM_TX_MODE. Received HMC field bits are passed to layer management by the signal, LM_RX_MODE. The application to hysteresis (reception of three identical values) significantly reduces the effects of link errors.

9.2.2.5 Template exchange notification bit—(Bit 1)

The template exchange notification (TXC) bit is a single-status bit employed in the HMC field to notify the peer HMC entity that it has effected a reconfiguration of channel bandwidth assignment as specified by the CCITT Q.93x management/signalling messages from the AU.

9.2.2.6 HMUX multiframing bit—(Bit 0)

The HMUX multiframing (HMF) bit is used to guard template exchange timing integrity. The HMF bit is asserted in the HMC field of every eighth TDM frame. It provides a multiframing pattern by which peer HMUX sublayers can synchronize their actions. The procedures for this multiframe operation are defined in 9.14 through 9.16.

9.2.3 HMUX modes

Four bearer service notification values have been defined. Together, these "modes" of bearer service permit a wide range of ISTE and AU functionality to be supported. The range of equipment spans rudimentary integrated services digital network (ISDN) basic rate only services, IEEE 802 only services, and equipment that supports a dynamic bandwidth management (including C channel services).

In the most simple network topology, the payload space of the TDM frame is entirely allocated to the P channel (Mode 0). This is the default configuration.

In the most encompassing network topology, the payload space may be mapped in a completely dynamic manner into multiple sets of channels, each bearing a different service and operating at a different transmission rate. When the layer management interface sends an LM_TX_MODE signal to the HMUX sublayer, the mode of operation is loaded into both the transmitter and receiver circuits.

9.2.3.1 HMC MODE interworking

The value in the MODE field identifies a particular bearer service that the ISTE is capable of supporting. Table 9-2 is an interworking matrix that describes combinations of equipment types and whether there is bearer service interworking across an IEEE 802.9 subnetwork. The exchange of MODE information between the AU and the ISTE establishes which forms of bearer service are possible.

Table 9-2—ISTE and AU mode interworking matrix

AU	ISTE				
	Mode 0 (802 only)	**Mode 1 (BRI-ISDN only)**	**Mode 2 (802 & BRI)**	**Mode 3 (Full dynamic)**	**Mode 4-7 (Reserved)**
0 (802 only)	802 services	No interworking	802 services	802 services	Reserved
1 (BRI-ISDN only)	No interworking	BRI services	BRI services	BRI services	Reserved
2 (802 & BRI)	802 services	BRI services	Both 802 & BRI	Both 802 & BRI	Reserved
3 (Full dynamic)	802 services	BRI services	Both 802 & BRI	Full dynamic	Reserved
4-7 (Reserved)	Reserved	Reserved	Reserved	Reserved	Reserved

9.2.3.2 ISTE and AU interworking descriptions

To ensure interworking, the mode of operation of peer HMUX sublayers must be compatible for both the transmit and the receive operations. This compatibility is ensured by setting the HMUX MODE register via the LME interface.

In situations wherein an ISTE or an AU is capable of flexible control of its operational modes, the actual operational mode that is configured may be a matter for negotiation over the D channel signalling service.

Reselection of mode operation over the IEEE 802.9 interface requires explicit management action.

9.2.3.2.1 Mode 0—IEEE 802 service only

This mode is used by IEEE 802.9 compatible terminal equipment and terminal adaptors (TE/TAs) that implement only IEEE 802 LAN services. The MODE bits are set to "000." The entire payload space of the TDM frame is allocated to the P channel.

The use of Mode 0 indicates that there is no support for any other bearer service over the TDM frame. This limitation includes the ISDN signalling service over the D channel. Consequently, there can be no reselection of mode of operation over the IEEE 802.9 interface, and the Mode 0 service is maintained via signal exchange across the IEEE 802.9 interface.

The use of Mode 0 implies that an HMUX is not required to perform dynamic bandwidth management functions. This mode permits the implementation of simple and economical IEEE 802 service only TEs/TAs.

9.2.3.2.2 Mode 1—BRI ISDN service only

This mode is used by IEEE 802.9 compatible ISTEs that implement only the basic rate ISDN interface services. The MODE bits are set to "001" indicating that there is no support for IEEE 802 services.

The use of Mode 1 implies that an HMUX is not required to support any form of dynamic bandwidth management. This mode permits the implementation of simple and economical BRI-ISDN TAs. In this mode, it is not necessary to implement an IEEE 802.9 MAC.

9.2.3.2.3 Mode 2—802 and BRI ISDN service only

This mode is used by IEEE 802.9 compatible ISTEs that implement only the IEEE 802 LAN services and basic rate ISDN services. The MODE bits are set to "010" indicating that there is no support for C channel services. Consequently, the entire payload space of the TDM frame is allocated to the P channel. The support for BRI-ISDN included in this mode implies that there is support for a D channel service. The use of Mode 2 implies that an HMUX is not required to support dynamic bandwidth management procedures for the C channel, as described in Chapter 11.

The use of this mode permits a minimal HMUX to be implemented. It does not require the HMUX multiframing nor the support of the dynamic bandwidth management procedures.

9.2.3.2.4 Mode 3—Dynamic bandwidth management service

This mode is used by IEEE 802.9 compatible ISTEs that implement both the IEEE 802 LAN services, and the ISDN basic rate ISDN interface services, and that can support an arbitrary number of additional wideband C channels. The MODE bits are set to "011." This mode requires more functionality to be capable of negotiating for the establishment and release of wideband channels.

The establishment and release of channels are subject to negotiation over the D channel in order to allocate the payload space, as described in Chapter 11.

9.2.3.2.5 Modes 4 through 7—Reserved

Modes 4 through 7 are reserved for future definition as operational bearer services.

9.2.4 PHY isochronous interface signal set

The HMUX higher layer interface provides a set of channels (P, AC, B1, B2, D, and multiple C channels) both to the IEEE 802.9 MAC Layer and to the isochronous services interface. The interface to the IEEE 802.9 MAC is defined in Chapter 8. The application uses of the isochronous bearer channels are beyond the scope of this standard. Consequently, multivendor implementations could easily become sufficiently varied as to create serious interworking problems. It is highly desirable, therefore, that there should be a common physical interface specification (input/output signals).

Adherence to the following subclauses will ensure that there is significant vendor independence over the HMUX (PHY) user interface. The set of logical signal specifications detailed below may usefully serve as a guide for implementation, and in fact, create the nucleus around which implementations are built.

The IEEE 802.9 PHY Layer offers an octet-based bearer service (see Chapter 7). Although the following interface specification is defined as bit serial, the TDM_Frame_Time signals provide a timing reference by which groups of eight contiguous serial bits can be aligned correctly as octets. The use of this bit serial interface is consistent with the serial conveyance of information within the PHY and with interface signal minimization. The grouping of eight serial bits into time slots is illustrated in table 9-3 and figure 9-6. The following is a description of the complete set of signals that constitutes a working HMUX sublayer to an application type of interface.

9.2.4.1 PHY_Active

This output signal asserts that the PHY Layer entity is operational and is able to engage in the reception/transmission of PDUs from Layer 2. This means that the PMD entity is operational, the PS sublayer is fully locked (synchronized), and the HMUX is properly functional.

9.2.4.2 Received_Data

This output signal represents a bit serial stream passing the entire TDM frame across the PHY interface.

9.2.4.3 Received_Clock

This output signal is the recovered clock signal that the PHY detects as part of the data information reception process. This is a continuous PMD recovered clock signal, at the IEEE 802.9 aggregate line rate. One edge of this clock signal is centrally located within the bit cell.

9.2.4.4 Received_TDM_Frame_Time

This output signal is active over a period of eight Receive_Clock signals that are aligned with the transfer of the Sync field information in the incoming Received_Data stream. This signal permits the isochronous user to obtain octet alignment of the received bit stream.

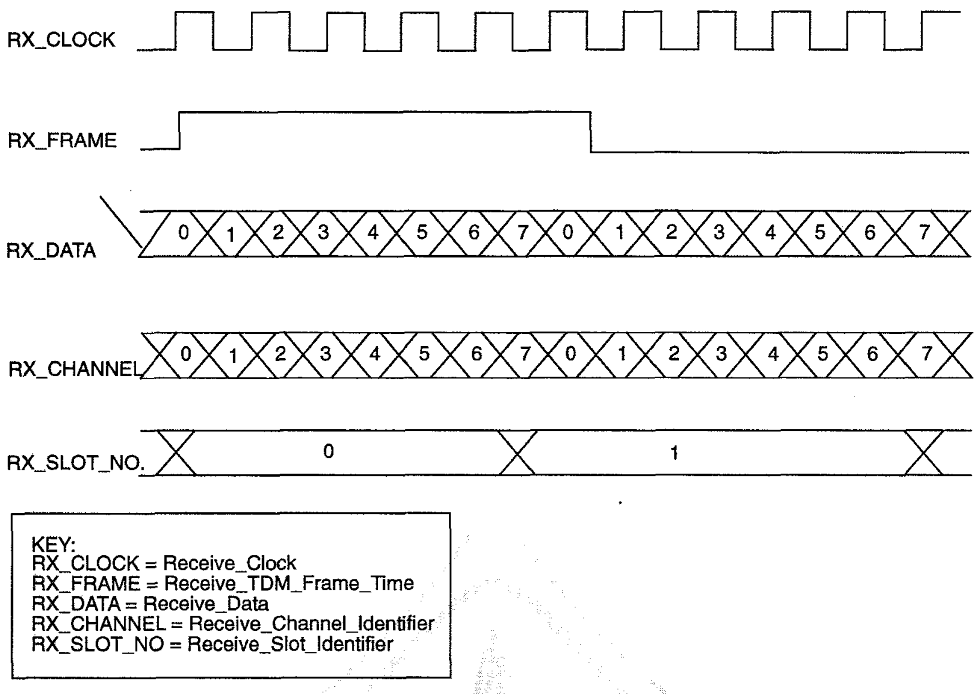


Figure 9-6—Timing relationships of received PHY signals

Table 9-3—PHY Layer signals

Signal name	AU	ISTE	No. of signals
PHY_Active	Output	Output	1
Receive_Data	Output	Output	1
Receive_Clock	Output	Output	1
Receive_TDM_Frame_Time	Output	Output	1
RX_Ch_ID_LSB	Output	Output	1
RX_Ch_Id_MSB	Output	Output	1
Receive_Invalid	Output	Output	1
Transmit_Data	Input	Input	1
Transmit_Clock	Input	Output	1
Transmit_TDM_Frame_Time	Input	Output	1
TX_Ch_Id_LSB	Output	Output	1
TX_Ch_Id_MSB	Output	Output	1

9.2.4.5 Received_Channel_Identifier

This pair (least and most significant bytes) of output signals provide a Channel_Identifier value presented in two separate binary sequentially-encoded octets. The Channel_Identifier identification is exactly aligned with the Receive_Data octet established by the Receive_TDM_Frame_Time signal. These two signals can therefore support up to 65 536 separate isochronous channels or PhSAPs. The relationship of these signals is shown in figure 9-6.

9.2.4.6 Received_Invalid

This output signal is aligned with the Receive_Data signal. This signal indicates whether the Receive_Data stream can be considered valid. The "validity" of the bit stream is influenced by the PMD, the PS, and the HMUX sublayer functions that monitor and inspect the error/synchronization aspects of the incoming bit stream. When this signal is asserted, the internal operations of the PHY sublayers may additionally cause the PHY-LM signals to be generated, but still allow the isochronous bearer service users to interpret the received data stream.

9.2.4.7 Transmit_Data

This input signal is offered to the PHY at the line rate, as determined by the Transmit_Clock signal. It must be offered to all isochronous service interface users, including any P, AC, B1, B2, D or C channels. The individual service bit streams must be offered to the PHY at the aggregate line transmission rate, as determined by the Transmit_Clock signal, and in accordance with the Transmit_Channel_Identifier signal (see 9.2.4.10).

9.2.4.8 Transmit_Clock

This signal may be used either as an input or an output. The use of this signal is dependent upon a PHY LME that controls the PHY Layer functionality and asserts the correct operation in both the AU and the TE/TA modes of operation. In the AU, this signal is an input that determines the link line rate and ensures that an ISTE can be synchronized to the host ISDN wide area network (WAN) to which the AU is attached. In the ISTE modes of PHY operation, this signal is an output signal that provides timing for Transmit_Data such that it is synchronous with the Receive_Clock signal. In this way, the ISTE isochronous user service is able to supply Transmit_Data at the correct times to ensure that the ISDN synchronous digital hierarchy requirements can be offered to the user of the IEEE 802.9 PHY.

9.2.4.9 Transmit_TDM_Frame_Time

This signal may be used either as an input or an output. The use of this signal is dependent upon a PHY LME that controls the PHY Layer functionality and asserts the correct operation in both the AU and the TE/TA modes of operation. In the case of an AU, this signal is an input that sets the IEEE 802.9 TDM frame alignment. It is required to permit proper allocation of channels within the PHY Layer, and to adhere to ISDN timing requirements. In the case of the ISTE modes of PHY operation, this signal is an output that provides the timing for alignment of the Transmit_Data to match the IEEE 802.9 TDM frame alignment.

9.2.4.10 Transmit_Channel_Identifier

This pair (least and most significant bytes) of output signals provides a Channel_Identifier value presented in two separate binary sequentially-encoded octets. The Channel_Identifier information provided by these signals specifies the Transmit_Data that should be presented to the HMUX. The timing relationship that should exist between the Transmit_Channel_Identifier signal and the presentation of the Transmit_Data is determined by the state of the loop TDM frame option (see 9.2.11). These two signals can support up to 65 536 separate isochronous channels or PhSAPs. The relationship of these signals is shown in figure 9-7.

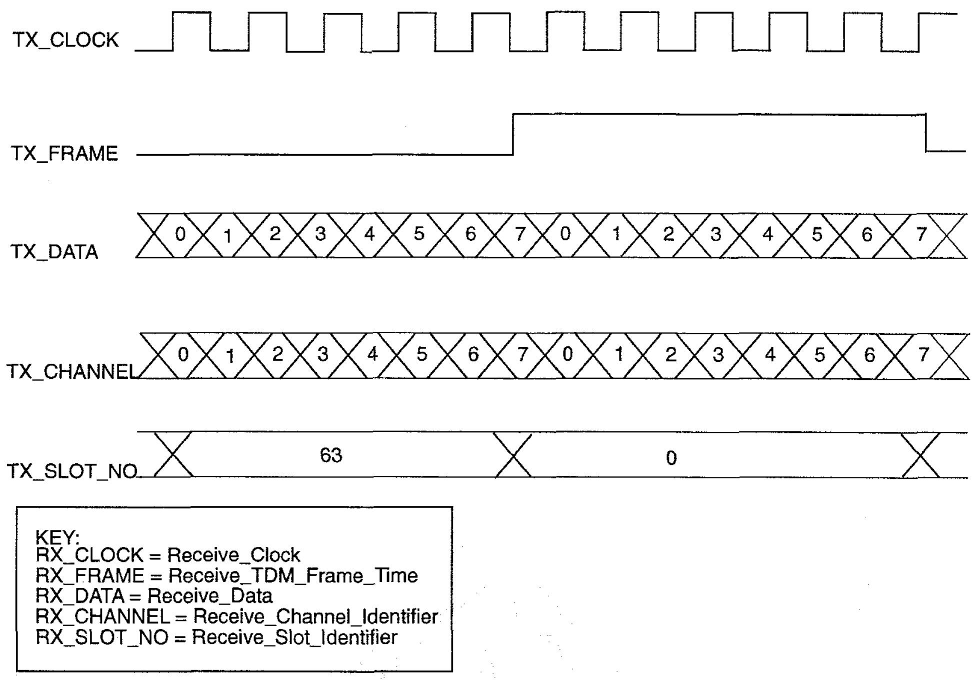


Figure 9-7—Timing relationship of transmitted PHY signals

At the ISTE in which the TDM_Loop_Delay has been specified as two octets, the timing relationship of the receive and transmit PHY interface is illustrated in figure 9-8. For a given TDM slot, the TX_Data is delayed by two octets relative to the RX_Data octet. Note that the RX_Channel_Id and the TX_Channel_Id signals are also delayed by two octets. This delay is an optional mechanism that is implementation dependent. It should be further noted that both figures 9-7 and 9-8 are shown for the situation where there is a 4.096 Mb/s PMD. Therefore the "slot numbers" are enumerated from 0 to 63.

9.2.5 HMUX sublayer initialization procedures

9.2.5.1 General

This subclause describes the initialization procedure for the HMUX sublayer. The HMC field is described in 9.2.1. In this model, the PHY Layer is in the powered-on condition. Therefore, the following procedure specifies the initialization procedure after power-up, the initialization of the PS sublayer (PS_LOCK), and the error recovery procedure for the PHY Layer connection. The HMUX sublayer executes a handshake procedure with the peer entity's HMUX sublayer as described in the finite state machine (FSM) in the following subclauses.

9.2.5.2 Transmitter initialization

After power-on, the default status of the HMUX transmitter is that it sends no multiframe signal, no TXC signal, nor any DCR information. The MODE is set to IEEE 802 only. Thus the HMC is sent as "00000000." On command by the LME, the HMUX transmitter executes in the mode of operation dependent upon the HMUX mode bits. The status of the DCR bit is also set by an exchange with the LME interface.

9.2.5.3 Receiver initialization

After power-on, the default status of the HMUX receiver is that it passes received HMC field information on toward layer management. But the receiver does not act upon any received TXC, MODE, or DCR signals

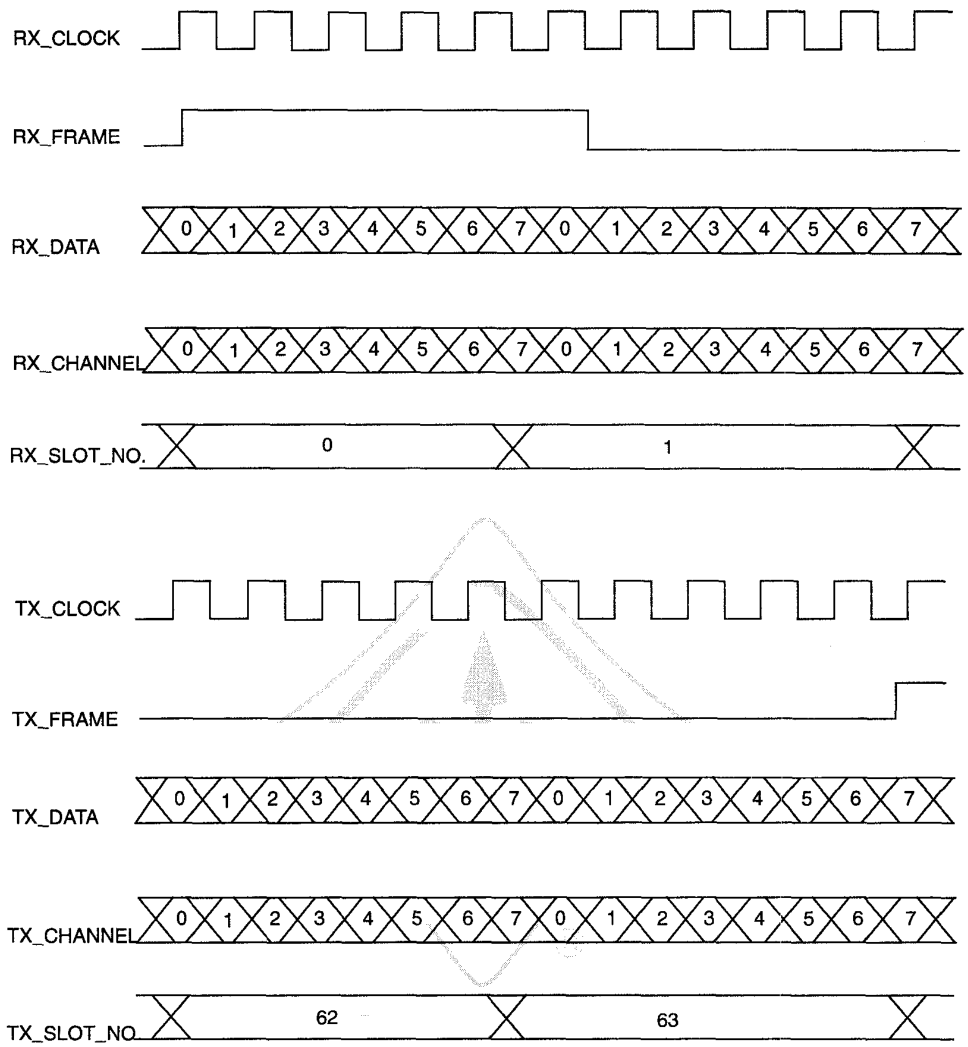


Figure 9-8—Timing relationship of PHY signals with loop TDM frame option

detected in the received HMC field. The HMUX receiver FSM immediately searches for HMUX multiframe lock by scanning the received HMC fields for the HMF bit. When the HMUX receiver detects this condition, it will assert the MF_LOCK signal. Upon command by the layer management, the HMUX receiver executes in the mode of operation that is selected by the HMUX mode bits. The status of the DCR bit is also signalled over the layer management interface toward the HMUX.

9.2.5.4 Definition of external signals and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this initialization procedure. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

9.2.5.4.1 External signals

These are the primitive signals that appear as external stimulus to the HMUX control FSM:

LM_PH_ACTION(activate) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are activated and the PHY Layer HMUX control state machine will enter the AWAIT_PS_LOCK state.

LM_PH_ACTION(deactivate) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are deactivated and the PHY Layer HMUX control state machine will enter the INACTIVE state.

LM_PH_ACTION(reset) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are initialized and the PHY Layer HMUX control state machine will enter the AWAIT_PS_LOCK state.

LM_TX_MODE — This is a specific management request for setting the HMUX mode. It is independent of the LM-PH-ACTION request. When this is issued, the HMUX control state machine will enter the correct mode of operation.

PS_LOCK — This signal from the PS sublayer indicates the status of the PS initialization (i.e., if it is in synchronization with the remote PS peer).

9.2.5.4.2 State names

These are the definitions of the states used in the functional model for the PHY initialization FSM:

INACTIVE — The INACTIVE state is the deactivated condition. The PHY HMUX control FSM will vector into the AWAITING_PS_LOCK state when it receives the LM-PH-ACTION(activate) signal (i.e., power-up). In this state, the FSM will not respond to any stimulus except for messages from its LME.

AWAITING_PS_LOCK — The AWAITING_PS_LOCK state is a state in which the HMUX sublayer is awaiting a signal from the PS sublayer indicating that it is synchronized with the remote PS sublayer. The PHY HMUX control FSM will vector into the SEEK_MF_LOCK state upon receipt of the PS_LOCK=1 signal if the mode does require multiframing (MODE>2). The PHY HMUX control FSM will vector into the MF_LOCK state upon receipt of the PS_LOCK=1 signal if the mode does not require multiframing (MODE<3).

SEEK_MF_LOCK — The SEEK_MF_LOCK is a state in which the HMUX sublayer is receiving a TDM frame with the multiframe pattern but the correct number of multiframe signals has not been received. The PHY HMUX control FSM will vector into the MF_LOCK state when it receives the RX_HMF_FAW=1 signal. If the FSM detects a loss of multiframing (RX_HMF_FAW=0) and it is operating in a mode that requires this (MODE>2), then it will vector into the AWAITING_PS_LOCK state.

MF_LOCK — The MF_LOCK state is the state in which the HMUX control sublayer FSM is in the frame synchronization condition and is receiving the multiframing signal (RX_HMF_FAW=1). If the FSM detects a loss of multiframing (RX_HMF_FAW=0) and it is operating in a mode that requires this (MODE>2), then it will vector into the AWAITING_PS_LOCK state. If the FSM detects a loss of multiframing (RX_HMF_FAW=0) and it is operating in a mode that does not require this (MODE<3), then it will stay in the MF_LOCK state.

9.2.5.4.3 Managed objects

The following MO within the FSM is controlled or "managed" by the management entity within the ISTE and AU:

RX_HMF_ERRORS — This managed 8-bit up/down counter is used as a hysteresis mechanism to monitor the multiframe signal from the remote peer. It is incremented upon receipt of an incorrect multiframe signal. It is decremented upon receipt of a good multiframe signal until it reaches a count of value "0."

9.2.5.5 FSM for HMUX control initialization

Figure 9-9 and tables 9-4a and 9-4b describe the FSM for the HMUX control initialization procedure.

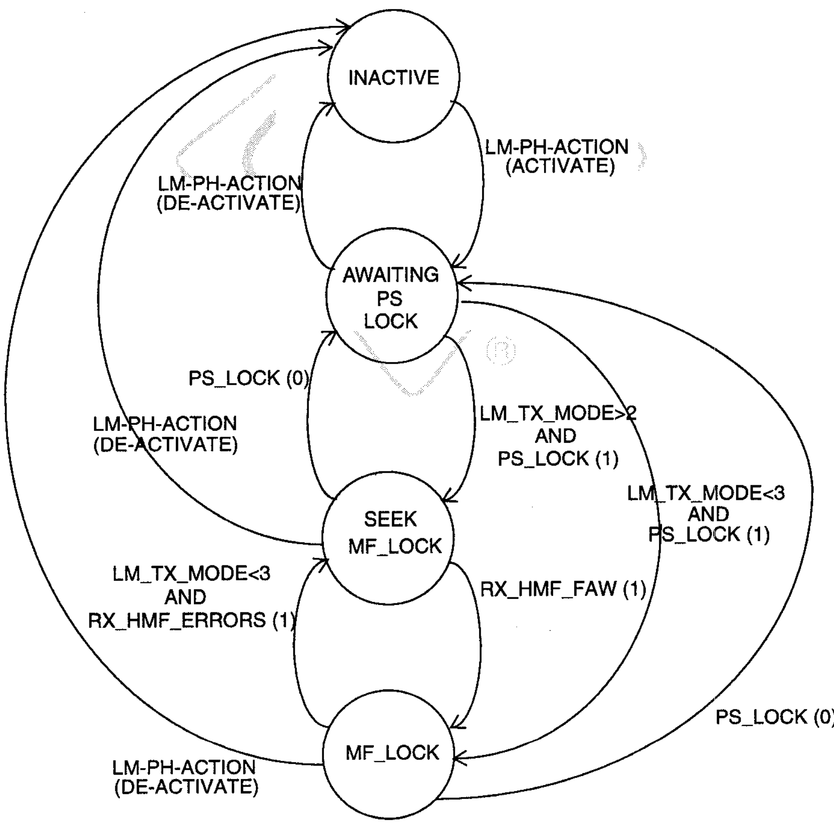


Figure 9-9—FSM state diagram for HMUX control initialization

Table 9-4a—FSM state table for HMUX control initialization for INACTIVE, AWAITING_PS_LOCK, and SEEK_MF_LOCK states

Current state	Event	Action(s)	Next state
INACTIVE	LM-PH-ACTION(activate)	SET_VARIABLES_ AND_ TIMERS SEND_HMC (0)	AWAITING_PS_ LOCK
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
AWAITING_PS_ LOCK	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AWAITING_PS_ LOCK
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (RESET)	AWAITING_PS_ LOCK
	PS_LOCK (1) and LM_TX_MODE>2	SEND_HMC (0)	SEEK_ MF_LOCK
	PS_LOCK (1) and LM_TX_MODE<3	SEND_HMC (hmc) LM-PH-NOTIFICATION (HMC_LOCK)	MF_LOCK
SEEK_MF_ LOCK	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	SEEK_ MF_LOCK
	LM-PH-ACTION(deactivate)	NO-ACTION	INACTIVE
	LM-PH-ACTION(deactivate)	SET_VARIABLES_AND_TIME RS_TO_DEFAULT	SEEK_ MF_LOCK
	PS_LOCK (0)	SEND_HMC (0)	AWAIT_ PS_LOCK

9.2.6 Dynamic link bandwidth allocation

In Mode 3, there is provision for dynamic allocation of channels in support of multiple concurrent services through the use of a hybrid template control mechanism. In this mode, a template descriptor is loaded into the background template register via the HMUX LME interface.

In Modes 0, 1, and 2 there is no requirement for the implementation of a hybrid template FSM in the HMUX sublayer. This permits minimal mode functionality to be built into terminal equipment (TE) because neither the HMUX template exchange FSM procedures nor the CCITT Q.93x based bandwidth management negotiation procedures are required.

In Mode 3, it is possible to dynamically reallocate the bandwidth of the payload field of the TDM frame through the use of the TXC control signal, the multiframing synchronization protection, and the internal organization of the HMUX channel multiplexer module. In order to effect changes in the allocation of channels (establishment and release), the TDM slot to channel mapping must be altered. To avoid service disruption to any given channel, it is essential that the reallocation of bandwidth take place without corrupting any channel that is already in use. The HMUX sublayer achieves this in Mode 3 operation through the use of the hybrid template functional module. The hybrid template functional module contains

Table 9-4b—FSM state table for HMUX control initialization for SEEK_MF_LOCK and MF_LOCK states

Current state	Event	Action(s)	Next state
SEEK MF_LOCK	RX_HMF_FAW (0)	SEND_HMC (0)	SEEK_ MF_LOCK
	RX_HMF_FAW (1)	SEND_HMC (hmc) LM-PH-NOTIFICATION (HMC_LOCK)	MF_LOCK
	LM-PH-ACTION (MODE<3)	SEND_HMC (hmc) LM-PH-NOTIFICATION (HMC_LOCK)	MF_LOCK
MF_LOCK	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	MF_LOCK
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (INVALID_COMMAND)	MF_LOCK
	LM-PH-ACTION (MODE<3)	SEND_HMC (hmc)	MF_LOCK
	PS_LOCK (0)	SEND_HMC (0) LM-PH-NOTIFICATION (HMC_UNLOCK)	AWAIT_ PS_LOCK
	LM_TX_MODE>2 and RX_HMF_ERRORS (0)	SEND_HMC (hmc)	MF_LOCK
	LM_TX_MODE<3 and RX_HMF_ERRORS (1)	SEND_HMC (0) LM-PH-NOTIFICATION (HMC_UNLOCK)	SEEK_ MF_LOCK

two stores for foreground and background descriptors of the PS slot to HMUX channel mapping. The foreground descriptor is used to control the (de)multiplexing of the information stream and the background descriptor is used as a temporary store for a new descriptor describing a new allocation of channel bandwidth. The structure of the descriptor registers may vary both according to the mode that has been selected and by the choice of channel capabilities designed into a given implementation. However, the architecture of the bandwidth allocation mechanism remains the same in all situations. Figure 9-10 illustrates the template exchange from background to foreground descriptor registers.

9.2.6.1 Bandwidth reallocation process

The mechanism of altering the allocation of bandwidth takes place in two phases:

— The background templates at both the transmitter and the receiver must be loaded with the same model of the slot to channel mapping.

— The exchanging of the background and the foreground templates at the transmitter and the receiver must be performed synchronously.

The first phase is effected by means of the extended CCITT Q.93x protocol exchange over the D channel of the IEEE 802.9 interface. These procedures and messages are described in Chapter 11 and annex D.

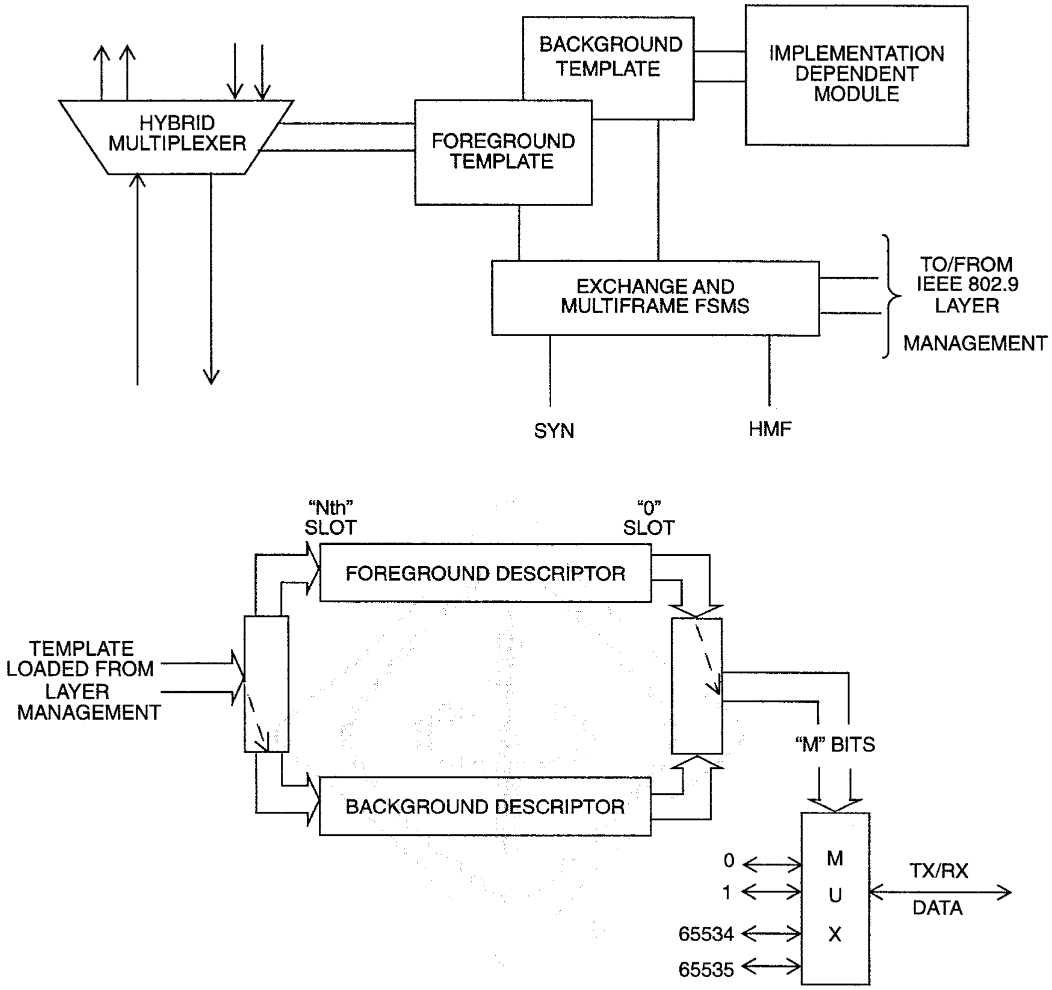


Figure 9-10—Foreground and background template descriptor exchange

In summary, the first phase is comprised of the following series of steps:

a) Use of CCITT Q.93x signalling protocol to establish the need for a change of link bandwidth allocation. This may be initiated either by an ISTE or the AU for the establishment/release of a channel.

b) Allocation by the AU bandwidth manager for a particular template (slot to channel mapping).

c) Transmission of this allocation assignment via the CCITT Q.93x extended channel identification and extended bearer capability information elements as part of the signalling SETUP message. This information is exchanged via the D channel.

d) Transformation of the contents of the CCITT Q.93x template description into implementation-specific HMUX layer management information.

e) The separate loading of this information by a layer management coordination function into both the ISTE HMUX sublayer and the AU HMUX sublayer.

f) (Optional) AU notification by an ISTE CCITT Q.93x message that the background loading task is complete.

Details of the relevant CCITT Q.93x messages may be found in annex D and in Chapter 11. The rules for the construction of the templates, as they are loaded into the background template store and then parsed by the foreground task, are described in 9.2.9.2.

The second phase requires that both an AU and its peer ISTE change over their background and foreground templates at identical points in the transmitted and received TDM frames. This demands a common timing reference for both the AU and the ISTE. This is under the functional control of the HMUX multiframing procedures detailed in 9.2.7 through 9.2.10.

9.2.6.2 Template structure rules

Rules for the construction of the templates, as they are loaded into the background template store and then subsequently parsed from the foreground template by the HMUX sublayer, are essential to permit implementation independent interworking.

The first eight octets of all TDM frames, independent of the HMUX operational mode, are statically defined. To support these there is no need for any form of foreground or background template storage. However, in the case of Mode 3, in which multiple channel allocations may be dynamically established, there are numerous channel-to-TDM slot mapping structures that may be defined to describe the organization of the remainder of the TDM frame, i.e., the "payload."

In the course of multiple C channel establishments and releases, a situation may arise in which it is no longer possible to allocate contiguous slots to fully establish the requested bandwidth. These dynamically established channels comprised of two or more slots can still be realized from two or more blocks on one or more contiguous slots which together provide all of the slots needed. The sequential (monotonic) ordering of slots within the TDM frame sequence must be maintained. This arrangement leads to the concept of channel fragments. Channel fragments are the disjoint sequentially ordered set of slots that together comprise the multislot (128 kb/s or more) bearer channel within the payload part of the TDM frame. Using this concept, the degree of flexibility that can be accommodated by a given template structure is bounded only by the maximum number of channel fragments supportable by the template descriptor. Note that when channel fragmentation occurs, there are more channel fragments than there are distinct channels.

The actual structure of the templates in the HMUX sublayer may take many forms. However, by the use of the channel fragmentation concept and adherence to the following rules, it is possible to achieve vendor independent interworking and a degree of implementation flexibility. The strict definition of the channel allocation messages at the CCITT Q.93x level may be transformed into implementation-dependent HMUX template store data structures in appropriate device drivers, without impairing vendor independence. Only one implementation-dependent parameter is required to achieve this. It is the Maximum_Number_Of_Fragments value.

During the CCITT Q.93x initialization procedures, there shall be a negotiation process in which the ISAU and the ISTE establish a maximum number of channel fragments that can be supported by particular interworking devices.

For this capability to be obtained, adherence to the following template structuring rules is essential:

a) Identical channel identifiers may be associated with one or more channel fragments.

b) Channel fragments are comprised of either one or a set of contiguous slots within the TDM frame.

c) Slots within channel fragments and channel fragments with the channel sequentially ordered constitute the complete channel.

d) An HMUX sublayer MO, Maximum_Number_Of_Fragments, shall be provided to identify the maximum number of channel segments that a given instance of the HMUX sublayer can support.

e) The minimum size of any channel fragment or channel shall be one octet (64 kb/s).

f) The maximum size of any channel fragment shall be 256 octets (16.384 Mb/s).

In line with the architectural constraint on the number of channels (65 536) that shall be supported, the maximum number of channel fragments is limited to 65 536.

9.2.7 HMUX multiframes

The HMUX multiframing technique predicts the time at which the next multiframe boundary will occur. It identifies a contiguous set of TDM frames that can be used as a frame of reference to ensure that errored TXC signals are correctly interpreted.

Common AU and ISTE timing is established by an HMUX specific TDM multiframe that is used to control the operation of template logic in the HMUX sublayer. The sequential delivery of the HMF bit in consecutive HMC fields constitutes a multiframe alignment word. The predictive qualities of the cyclically occurring HMUX multiframe signals are used to protect the precise timing of template exchange operations against the effects of receive PMD and link errors.

The HMUX multiframe (HMF) bit in the HMC field is distinct from bit 7 of the SYN field (see 6.5.3). It is used to signal the occurrence of an HMUX multiframe boundary. The HMF bit drives a multiframe FSM which synchronizes peer-to-peer HMUX operations over the IEEE 802.9 interface.

When enabled by the HMUX's layer management, the transmit HMF FSM asserts HMF=0 for seven consecutive TDM frames and the HMF bit is set to a "1" on every eighth TDM frame. If the HMUX's LME does not enable the HMF FSM, the HMF bit =1 signals will not be sent.

The receive HMC FSM, when enabled by the HMUX LME, acquires and maintains the HMUX multiframe lock based on the use of the HMC field HMF bit and hysteresis built into the HMF FSM. The RX_HMF FSM searches in the sequence of received HMF bits (bit "0" of the HMC field) for the frame alignment pattern, "100000001." Subject to detecting this pattern, HMF_LOCK state is established.

The HMF_LOCK state is lost when there has been an accumulation of three error bits in the repeated HMF sequence of "10000000." The running total of errors is accumulated by counting the detection of received HMF bits that are not consistent with the cyclic "10000000" pattern, with every correctly received value decrementing the running error count until the count reaches value "0." This hysteresis mechanism tolerates infrequent HMF receiver bit errors without loss of HMF_LOCK status. The HMF_LOCK state is lost when the PS sublayer Frame_lock state is lost. This will be the most probable and usual cause of losing HMC_LOCK status.

Note that the ISTE aligns its TX-HMF=1, and thus its transmit HMUX multiframe with the receipt of the AU HMF bit. This ensures that it is possible to exchange the ISTE templates synchronously in both the send and receive directions of the interface.

9.2.8 HMUX transmit multiframe FSM

9.2.8.1 General

An AU in Mode 3 operation transmits the HMF FAW "10000000" bit by bit in the HMF subfield of the HMC field.

While an AU generates the timing for transmission of the HMUX multiframe FAW, the ISTE synchronizes the transmission of its TX_HMF_FAW with timing obtained from the RX_HMF FAW sent by the AU.

9.2.8.2 Definition of external signals and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this HMUX control initialization procedure. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

9.2.8.2.1 External signals

The following primitive signals appear as external stimulus to the HMUX transmit multiframe FSM:

LM_PH_ACTION(activate)	When this management request is issued, the PMD, the PS, and the HMUX sublayers are activated.
LM_PH_ACTION(deactivate)	When this management request is issued, the PMD, the PS, and the HMUX sublayers are deactivated and the PHY Layer HMUX control state machine will enter the INACTIVE state.
LM_PH_ACTION(reset)	When this management request is issued, the PMD, the PS, and the HMUX sublayers are initialized and the PHY Layer HMUX control state machine will stay in the same state.
LM_TX_MODE	When this management request is issued, the HMUX control state machine will enter the correct mode of operation.
PS_LOCK	This is a signal from the PS sublayer indicating the status of the PS initialization (i.e., if it is in synchronization with the remote PS peer).
AU_MODE(0)	This is the mode of operation that the layer management has authorized. This designates IEEE 802 service only.
AU_MODE(1)	This is the mode of operation that the layer management has authorized. This designates BRI ISDN service only.

9.2.8.2.2 State names

The following are definitions of the states used in the functional model for the HMUX transmit multiframe FSM:

INACTIVE	The INACTIVE state is the deactivated condition. The PHY HMUX transmit multiframe FSM will vector into the AU_SEND state when it receives the LM-PH-ACTION(activate) signal (i.e., power-up) combined with the AU_MODE=1 (signifying that this device is the AU), and the layer management signal, LM_TX_MODE>2. For these same conditions but with the AU_MODE=0, the FSM will vector into the TE_AWAIT_MF state.
AU_SEND_MF	The AU_SEND_MF state is a state in which the AU's HMUX transmit multiframe FSM is sending the multiframing pattern. In this state, there is a recirculating 3-bit binary up counter, FR_CNT, that is used to keep track of the state of the HMF bit. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.
TE_AWAIT_MF	The TE_AWAIT_MF is a state in which the ISTE's HMUX transmit multiframe FSM is awaiting receipt of a synchronization signal,

PS_LOCK(1), from the PS sublayer to signal that the PS sublayer has achieved synchronization with its remote PS peer. Upon notification of this event, the FSM will vector to the TE_SEND_MF state. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

TE_SEND_MF — The TE_SEND_MF state is a state in which the ISTE's HMUX transmit multiframe FSM is sending the multiframing pattern. In this state, there is a recirculating 3-bit binary up counter, FR_CNT, that is used to keep track of the state of the HMF bit. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

9.2.8.2.3 Managed objects

The following MO within the FSM is controlled or "managed" by the management entity within the ISTE and AU:

FR_CNT — This resettable recirculating 3-bit binary counter is used to keep track of the state of the HMF bit for transmission.

9.2.8.3 FSM for HMUX transmit multiframe

Figure 9-11 and tables 9-5a and 9-5b describe the FSM for the HMUX transmit multiframe.

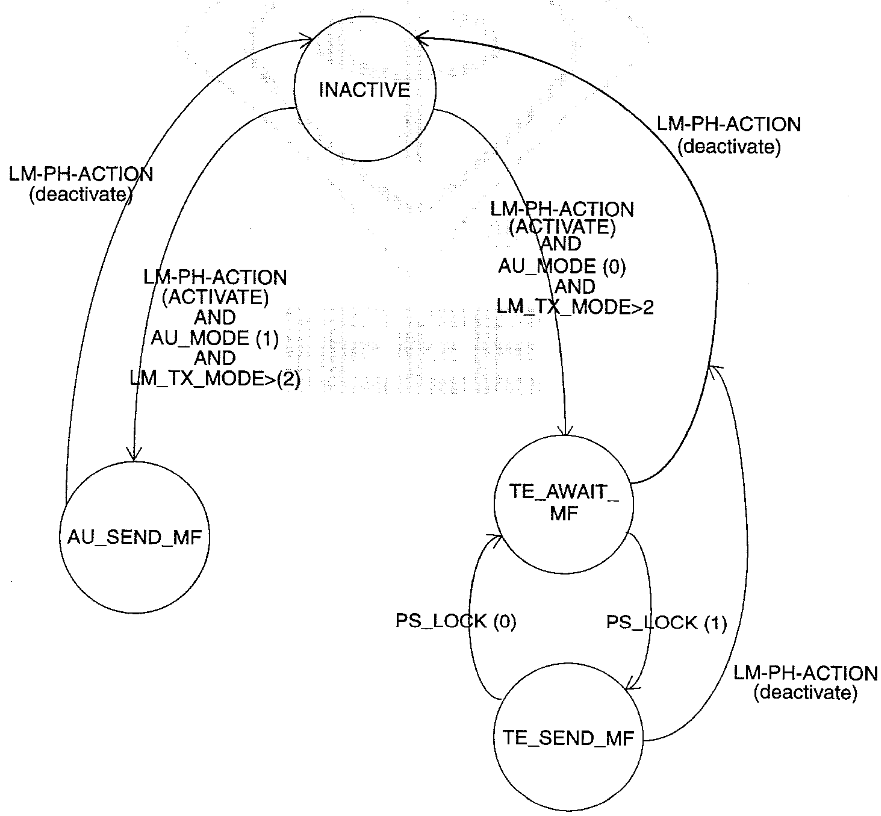


Figure 9-11—FSM state diagram for HMUX transmit multiframe

Table 9-5a—FSM state table for HMUX transmit multiframe for INACTIVE and AU_SEND_MF states

Current state	Event	Action(s)	Next state
INACTIVE	LM-PH-ACTION(activate) and LM_TX_MODE> (2) and AU_MODE(0)	SET_VARIABLES_AND_TIMERS SEND_HMF=0 SET_FR_CNT=0	TE_AWAIT_MF
	LM-PH-ACTION(activate) and LM_TX_MODE>(2) and AU_MODE (1)	SET_VARIABLES_AND_TIMERS SEND_HMF=1 SET_FR_CNT=0	AU_SEND_MF
	LM-PH-ACTION (deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
AU_SEND_MF	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AU_SEND_MF
	LM-PH-ACTION(deactivate)	SEND_HMF=0	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (RESET)	AU_SEND_MF
	FR_CNT≠7	SET_FR_CNT=FR_CNT+1 SEND_HMF=0	AU_SEND_MF
	FR_CNT=7	SET_FR_CNT=0 SEND_HMF=1	AU_SEND_MF

Figure 9-5b—FSM state table for HMUX transmit multiframe for TE_AWAIT_MF and TE_SEND_MF states

Current state	Event	Action(s)	Nest state
TE_AWAIT_ MF	LM_PH-ACTION (activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	TE_AWAIT_MF
	LM-PH-ACTION(deactivate)	SEND_HMF=0	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	TE_AWAIT_MF
	PS_LOCK (0)	SET_FR_CNT=0 SEND_HMF (0)	TE_AWAIT_MF
	PS_LOCK (1)	SET_FR_CNT=FR_CNT+1 SEND_HMF (1)	TE_SEND_MF
TE_SEND_MF	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	TE_SEND_MF
	LM-PH_ACTION(deactivate)	SEND_HMF (0)	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (INVALID_COMMAND)	TE_SEND_MF
	PS_LOCK (0)	SET_FR_CNT=0 SEND_HMF (0)	TE_AWAIT_MF
	FR_CNT≠7	SET_FR_CNT=FR_CNT+1 SEND_HMF (1)	TE_SEND_MF
	FR_CNT=7	SET_FR_CNT=0 SEND_HMF (1)	TE_SEND_MF

9.2.9 HMUX received multiframe FSM

9.2.9.1 General

The receive multiframe FSM is driven by the reception of the HMC field of the TDM physical frame. It operates on the HMF bit in the HMC field of the IEEE 802.9 TDM frame. The operation is identical for both the AU and the ISTE. The output of this FSM is used by both the HMUX transmit multiframe FSM and the HMUX control FSM.

When activated by the LM-PH-ACTION(activate) signal, the receive multiframe FSM enters the SEEK_FAW state and searches for a correct incoming frame alignment pattern (FAW=100000001) in the HMF bit of the received HMC field. Upon detection of this pattern, the HMUX receive multiframe FSM signals the HMUX multiframe lock condition by activating the LM_MF_LOCK signal to layer management. The receive multiframe FSM then vectors to the MF_ACTIVE state where it continues to monitor the HMF bit stream for proper frame alignment. If frame alignment is lost, the receive multiframe FSM vectors back to the SEEK_FAW state where it begins searching for proper frame alignment.

9.2.9.2 Definition of external signals and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this initialization procedure. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

9.2.9.2.1 External signals

The following primitive signals appear as external stimulus to the HMUX receive HMF FSM:

LM_PH_ACTION(activate) When this management request is issued, the PMD, the PS, and the HMUX sublayers are activated and the PHY Layer HMUX control state machine will enter the SEEK_FAW state.

LM_PH_ACTION(deactivate) When this management request is issued, the PMD, the PS, and the HMUX sublayers are deactivated and the PHY Layer HMUX control state machine will enter the INACTIVE state.

LM_PH_ACTION(reset) When this management request is issued, the PMD, the PS, and the HMUX sublayers are initialized and the PHY Layer HMUX control state machine will stay in the same state. This is a single-bit stream signal from the received HMC field's HMF bit.

RX_HMF This is delivered once every TDM frame period (frame period is 125 μs).

9.2.9.2.2 State names

The following are definitions of the states used in the functional model for the PHY initialization FSM:

INACTIVE The INACTIVE state is the deactivated condition. The PHY HMUX receive multiframe FSM will vector into the SEEK_FAW state when it receives the LM-PH-ACTION(activate) signal (i.e., power-up).

SEEK_FAW The SEEK_FAW state is a state in which the AU's HMUX receive multiframe FSM is searching for the reception of a valid FAW (FAW=100000001). Upon receipt of a valid FAW, the receive multiframe FSM will set the signal LM_MF_LOCK=1 to notify layer management that multiframe lock has been reached. The

FSM will then vector to the state MF_ACTIVE. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

MF_ACTIVE — The MF_ACTIVE state is a state in which the HMUX receive multiframe FSM is searching for a continued good pattern of frame alignment. In this state there is a resettable up/down 2-bit binary counter, MF_ERR_CNT, that is used to keep track of the reception of bad frame alignment patterns. Upon receipt of three bad multiframe patterns, the FSM will vector back to the SEEK_FAW state. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

9.2.9.2.3 Managed objects

The following MO within the FSM is controlled or "managed" by the management entity within the ISTE and AU:

MF_ERR_CNT — This resettable up/down 2-bit binary counter is used to keep track of the state of the HMF bit for keeping account of the multiframe frame alignment word.

9.2.9.3 FSM for HMUX received multiframe

Figure 9-12 and tables 9-6a and 9-6b describe the FSM for the HMUX receive multiframe.

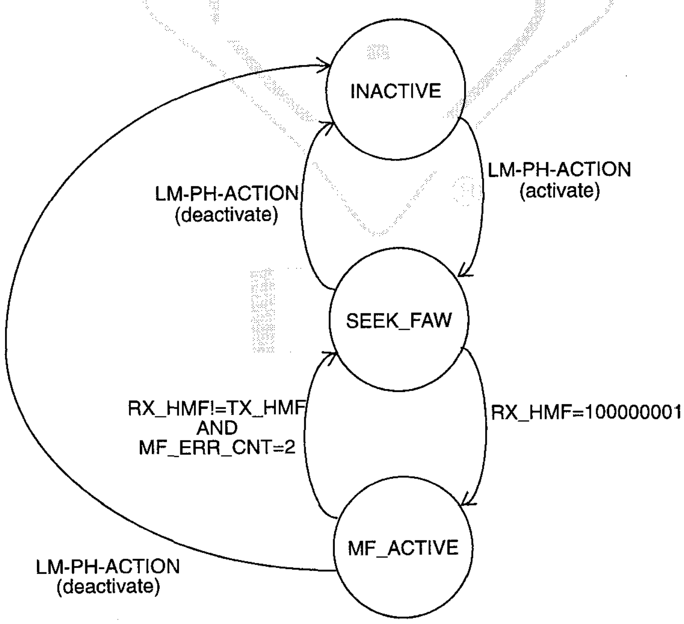


Figure 9-12—FSM state diagram for HMUX receive multiframe

Table 9-6a—FSM state table for HMUX receive multiframe for INACTIVE and SEEK_FAW states

Current state	Event	Action(s)	Next state
INACTIVE	LM-PH-ACTION(activate)	SET_VARIABLES_AND_TIMERS LM_MF_LOCK=0	SEEK_FAW
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
SEEK_FAW	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	SEEK_FAW
	LM-PH-ACTION(deactivate)	LM_MF_LOCK=0	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (RESET)	SEEK_FAW
	RX_HMF=100000001	LM-PH-NOTIFICATION (MF_ACTIVE) LM_MF_LOCK=1 MF_ERR_CNT=0	MF_ACTIVE

Table 9-6b—FSM state table for HMUX receive multiframe for MF_ACTIVE state

Current state	Event	Action(s)	Next state
MF_ACTIVE	LM-PHY-ACTION(activate)	LM-MA-NOTIFICATION (INVALID_COMMAND)	MF_ACTIVE
	LM-PHY-ACTION(deactivate)	LM_MF_LOCK=0	INACTIVE
	LM-PHY-ACTION(reset)	LM-MA-NOTIFICATION (INVALID_COMMAND)	MF_ACTIVE
	RX_HMF=TX_HMF and MF_ERR_CNT!=0	MF_ERR_CNT=MF_ERR_CNT-1	MF_ACTIVE
	RX_HMF!=TX_HMF and MF_ERR_CNT=2	MF_ERR_CNT=MF_ERR_CNT+1	MF_ACTIVE
	RX_HMF!=TX_HMF and MF_ERR_CNT=2[a]	LM_MF_LOCK=0	SEEK_FAW

[a]This event is reached when there is an accumulated error count of 2, and a third multiframe alignment error is reached.

9.2.10 HMUX template exchange FSM

9.2.10.1 General

The template exchange notification bit (TXC) is conveyed across the IEEE 802.9 interface via the TDM frame. This bit is normally transmitted as a "0" by both the ISTE and the AU. In order that a reallocation of payload field can be negotiated, AU layer management must issue the LM_TX_TXC signal to effect the following sequence of actions:

a) The AU issues a layer management command via the signal LM_TX_TXC to its HMUX sublayer.

b) The AU's HMUX sublayer awaits the commencement of the next HMUX transmit multiframe event.

c) The AU's HMUX sublayer sends a TXC=1 signal in each of the eight TDM frames contained within one HMUX multiframe period. The ISTE will return the received TXC signal back toward the AU in its next available HMC field.

d) Subject to link errors, the ISTE's HMUX sublayer receives eight TXC=1 signals in the received HMC field during this HMUX multiframe period. Upon the occurrence of the TDM SYN field in the ninth incoming TDM frame, the ISTE exchanges its foreground template with its background template.

e) The AU's HMUX sublayer monitors the received TXC bit in the received HMC field, and subject to an accumulated total of six TXC=1 bits in the received HMC stream, and the reception of the TDM SYN field in the ninth TDM frame, the HMUX will exchange its foreground template with its background template. The significance of the ninth frame is to ensure that there is multiframe alignment. Note that the multiframe is determined over a sequence of eight frame periods.

The effect of link errors that may corrupt the value of the TXC bits in the HMC field are reduced by the hysteresis of six "good" TXC bits. If there are five or less TXC=1 bits in any HMUX multiframe period, the receiving ISTE shall not exchange templates. The AU is able to detect the probable failure to exchange templates at the ISTE due to a lack of response from the ISTE.

The process of AU-to-ISTE and ISTE-to-AU template exchange may be seen as a symmetrical process. However, the above description is clearly asymmetric. This has the advantage that there is a single, full-duplex template (foreground/background pair) in the AU and also in the ISTE. If a symmetrical arrangement is required then there is a doubling of the amount of template storage and an increase in the amount of CCITT Q.93x message exchange required.

If an asymmetric scheme is deployed, thereby providing savings in terms of template storage and message processing, it is vital to have a TDM frame timing loopback (see 9.2.11). This will ensure that the total loop delay from AU-to-ISTE and back from ISTE-to-AU does not exceed eight TDM frames slots.

9.2.10.2 Definition of external signals, internal signals, and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this initialization procedure. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

9.2.10.2.1 External signals

The following primitive signals appear as external stimulus to the HMUX template exchange FSM:

AU_MODE	When this management request is issued, the HMUX sublayer is notified as to whether it is to operate in an AU mode or in an ISTE mode.
LM_PH_ACTION(activate)	When this management request is issued, the PMD, the PS, and the HMUX sublayers are activated and the PHY Layer HMUX control state machine will enter the AWAIT_PS_LOCK state.
LM_PH_ACTION(deactivate)	When this management request is issued, the PMD, the PS, and the HMUX sublayers are deactivated and the PHY Layer HMUX control state machine will enter the INACTIVE state.

LM_PH_ACTION(reset) When this management request is issued, the PMD, the PS, and the HMUX sublayers are initialized and the PHY Layer HMUX control state machine will stay in the same state.

LM_TX_MODE When this management request is issued, the HMUX control state machine will enter the correct mode of operation.

LM_TX_TXC When this management request is issued, the AU's HMUX is instructed to send the template exchange control bit.

PS_LOCK This signal from the PS sublayer indicates the status of the PS initialization (i.e., if it is in synchronization with the remote PS peer).

RX_MF This internal signal communicates the state of the received HMF bit from the incoming HMC field.

RX_TXC This internal signal communicates the state of the received TXC bit from the incoming HMC field.

START_OF_MF When this signal from the HMUX internal module is issued, the FSM is alerted that there is a start of a new multiframe period.

9.2.10.2.2 State names

The following are definitions of the states used in the functional model for the PHY initialization FSM:

INACTIVE The INACTIVE state is the deactivated condition. The PHY HMUX template exchange FSM will vector into the AU_AWAIT_TXC state when it receives the LM-PH-ACTION(activate) signal (i.e., power-up) combined with the AU_MODE=1 (signifying that this device is the AU), and the layer management signal, LM_TX_MODE>2. For these same conditions but with the AU_MODE=0, the FSM will vector into the TE_AWAIT_TXC state.

AU_AWAIT_TXC The AU_AWAIT_TXC state is a state in which the AU's HMUX template exchange FSM is awaiting the proper conditions before transmitting the template exchange control information as part of a template exchange synchronization handshake. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

AU_SEND_TXC The AU_SEND_TXC state is a state in which the AU's HMUX template exchange FSM is sending the template exchange control information. In this state, there is a 3-bit binary up counter, TXC_CNT, that is used to keep track of the state of the TXC bit. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

TE_AWAIT_TXC The TE_AWAIT_TXC state is a state in which the ISTE's HMUX template exchange FSM is awaiting the proper conditions before transmitting the template exchange control information as part of a template exchange synchronization handshake. Upon receipt of the LM-PH-ACTION(deactivate) signal (i.e., power down), the FSM will vector to the INACTIVE state.

9.2.10.2.3 Managed objects

The following MO within the FSM is controlled or "managed" by the management entity within the ISTE and AU:

TX_CNT	This resettable 3-bit binary up counter is used to keep track of the state of the TXC bit for transmission.

9.2.10.3 FSM for HMUX template exchange

Figure 9-13 and tables 9-7a, 9-7b, and 9-7c describe the FSM for the HMUX template exchange.

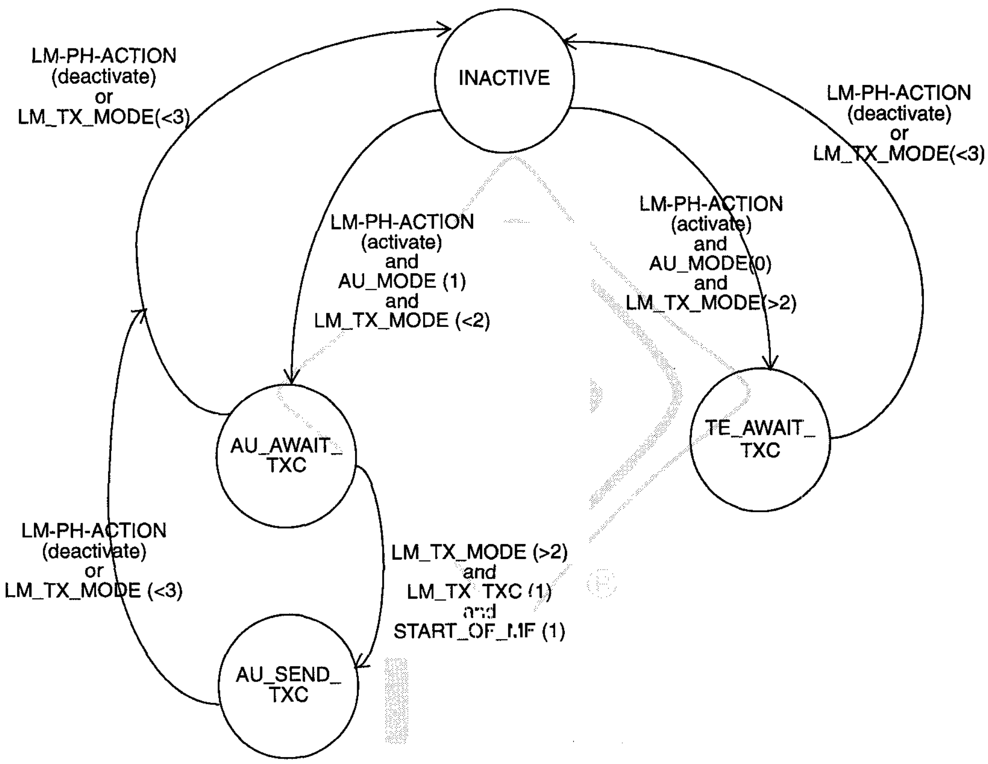


Figure 9-13—FSM diagram for HMUX template exchange

9.2.11 Loop TDM frame option

The uncertainties of the delays associated with the ISTE and the AU interworking functions, that arise from an arbitrary phase relationship of the transmit and receive TDM frames at an ISTE, impose some constraints on interworking and cost effective designs.

IEEE 802.9 channel allocations are symmetrical and full-duplex. This is advantageous because only a single template descriptor is required for the multiplexing functionality on the transmit and receive information streams. This economy of template storage buffers can also help reduce the layer management interface signals required.

Providing that there is a fixed slot (two-octet) delay between the transmit ISTE TDM frame and the TDM frame that it receives, the receive channel identifier is identical to the transmit channel identifier.

Table 9-7a—FSM state table for HMUX template exchange for INACTIVE and AU_AWAIT_TXC states

Current state	Event	Action(s)	Next state
INACTIVE	LM-PH-ACTION(activate) and LM_TX_MODE (>2) and AU_MODE (0)	SET_VARIABLES_AND_TIMERS SEND_TXC (0) HMUX_TXC=0 TXC_CNT=0	TE_AWAIT_ TXC
	LM-PH-ACTION(activate) and LM_TX_MODE (>2) and AU_MODE (1)	SET_VARIABLES_AND_TIMERS SEND_TXC (0) HMUX_TXC=0	AU_AWAIT_ TXC
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
AU_AWAIT_ TXC	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AU_AWAIT_ TXC
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (RESET)	AU_AWAIT_ TXC
	LM_TX_MODE (<3)	NO_ACTION	INACTIVE
	PS_LOCK (0)	NO_ACTION	INACTIVE
	LM_TX_MODE (>2) and LM_TX_TXC (1) and START_OF_MF (0)	SEND_TXC (0) HMUX_TXC=0	AU_AWAIT_ TXC
	LM_TX_MODE (>2) and LM_TX_TXC (1) and START_OF_MF (0)	SEND_TXC (1) HMUX_TXC=0	AU_SEND_TXC

Furthermore, it is advantageous at the PHY service interface to receive data that is accompanied by the receive channel identifier, while passing the transmit channel identifier ahead of the data that is to be transmitted.

Note that the fixing of the ISTE delay to two slots can ensure that the total AU-to-AU delay does not exceed eight slots.

9.3 PS sublayer

Figure 9-14 illustrates the detailed model of the transmitter for the PS sublayer. Figure 9-15 illustrates the detailed model of the receiver for the PS sublayer. In the transmit direction, the reserved field and the TDM MTN field are appended to the multiplexed data received from the HMUX sublayer. Then, through a gating function (which may cause the output of the PS to be set to the logic "1" level as required for initialization and maintenance procedures), data is acted upon by the scrambler function. The parity calculation is performed upon the scrambled data. The output of the parity calculation (the parity bits) are appended to the scrambled data stream. Finally, the SYN field is appended to constitute a complete TDM frame. This frame is sent on to the PMD sublayer for transmission.

The functional block diagram for the PS sublayer transmitter shows an input control signal, TX_TDM_VALID, serving as a gating function on the bit stream leaving the top MUX module. This reflects

Table 9-7b—FSM state table for HMUX template exchange for AU_SEND_TXC and TE_AWAIT_TXC states

Current state	Event	Action(s)	Next state
AU_SEND_TXC	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AU_SEND_TXC
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AU_SEND_TXC
	LM_TX_MODE (<3)	NO_ACTION	INACTIVE
	PS_LOCK (0)	NO_ACTION	INACTIVE
	START_OF_MF (0)	SEND_TXC (1)	AU_SEND_TXC
	START_OF_MF (1)	SEND_TXC (0) HMUX_TXC=1 LM-PH-NOTIFICATION (hmc_exchange)	AU_SEND_TXC
TE_AWAIT_TXC	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	TE_AWAIT_TXC
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (INVALID_COMMAND)	TE_AWAIT_TXC
	LM_TX_MODE (<3)	NO_ACTION	INACTIVE
	PS_LOCK (0)	NO_ACTION	INACTIVE

Table 9-7c—FSM state table for HMUX template exchange for TE_AWAIT_TXC state

Current state	Event	Action(s)	Next state
TE_AWAIT_TXC	START_OF_MF (0) and RX_TXC (0)	SEND_TXC (rx_txc)	TE_AWAIT_TXC
	START_OF_MF (0) and RX_TXC (1)	SEND_TXC (rx_txc) TXC_CNT=TXC_CNT+1	TE_AWAIT_TXC
	START_OF_MF (1) and RX_MF_BIT (0) and TXC_CNT (<6)	SEND_TXC (rx_txc) TXC_CNT=0 HMUX_TXC=0	TE_AWAIT_TXC
	START_OF_MF (1) and RX_MF_BIT (1) and TXC_CNT (>5)	SEND_TXC (rx_txc) TXC_CNT=0 HMUX_TXC=1	TE_AWAIT_TXC

the ability to choke or disable the output bit stream. The input control signal, PARITY DIAGNOSTIC, enters the parity generator and serves to force a diagnostic pattern into the transmitted bit stream. Both of these signals are an implementation extension.

In the receive direction, TDM framing is performed upon the incoming data stream as delivered from the PMD sublayer. First, the receiver must achieve clock timing synchronization by detecting the pattern in the

SYN field of the TDM frame. Next the parity is checked. If there is a parity error, the management entity is informed. The data is then sent on to the descrambler. From there, the received data is demultiplexed so that the TDM MTN and the reserved field information may be reported to management. The remaining data stream is gated by RX_TDM_VALID and transferred to the HMUX sublayer.

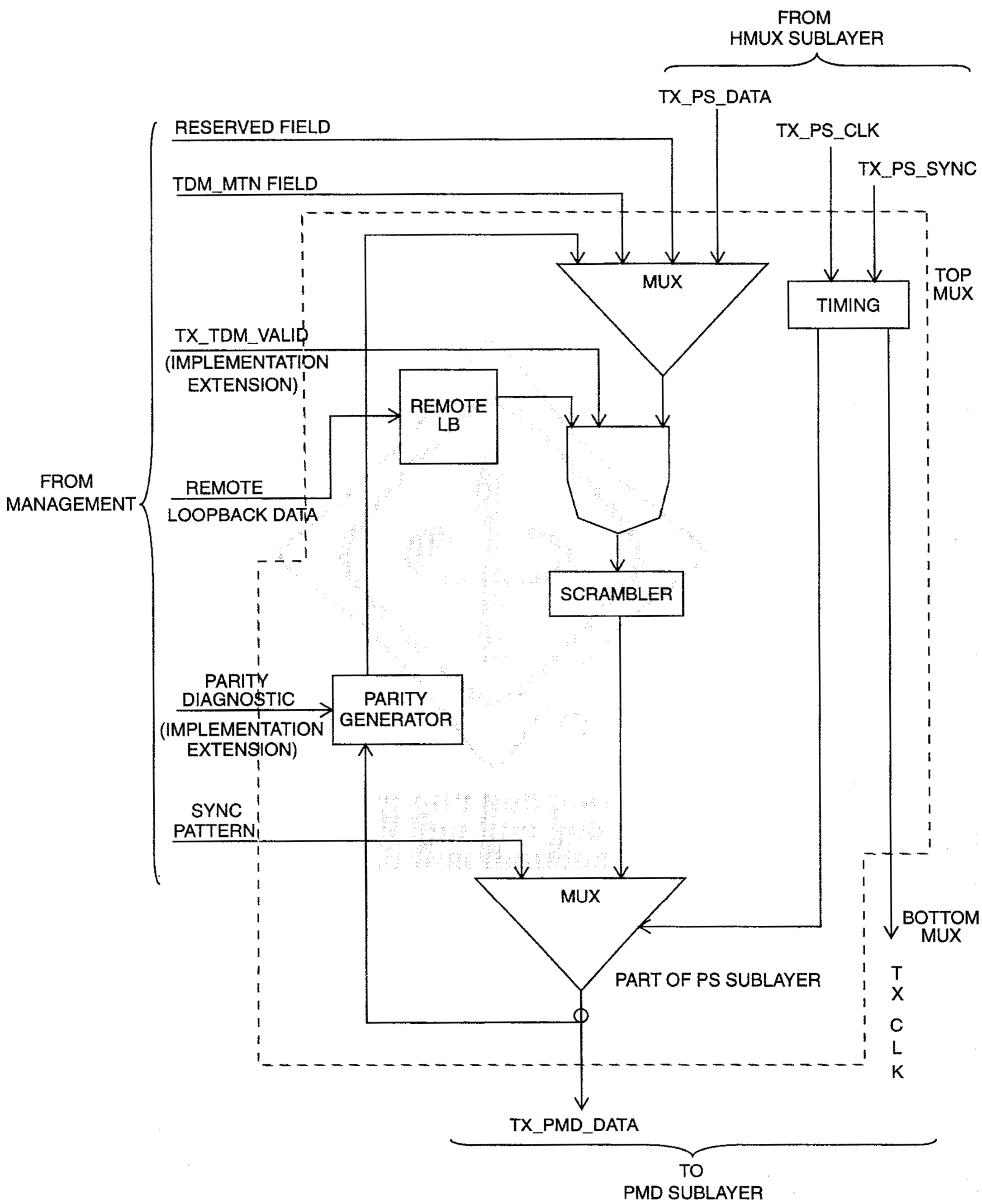


Figure 9-14—Detailed model of the transmitter for the PS sublayer

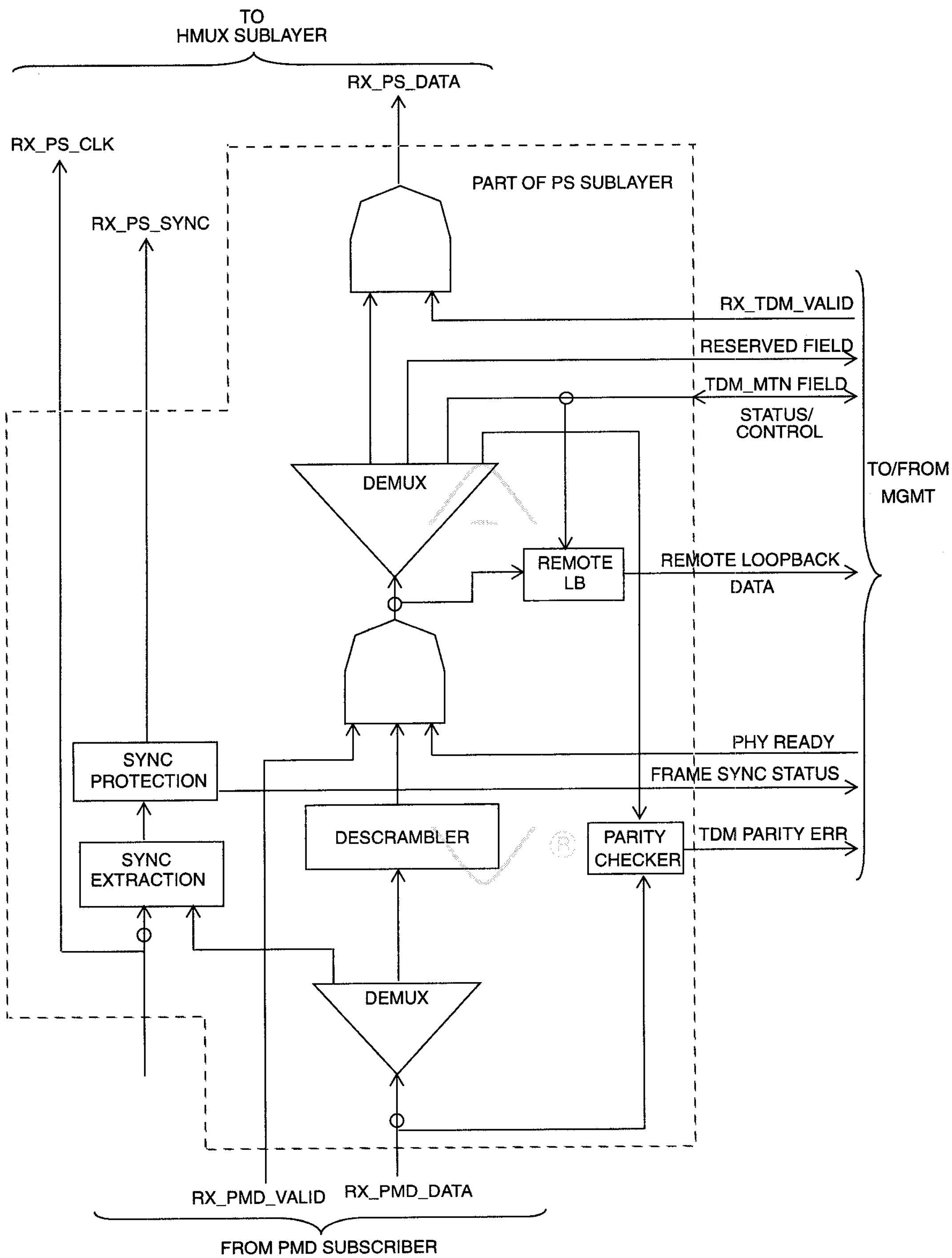


Figure 9-15—Detailed model of the receiver for the PS sublayer

9.3.1 Frame synchronization

Framing shall be performed based on the synchronization pattern provided within the SYN field of the TDM frame. This SYN field occupies the first octet of the TDM frame. The synchronization pattern is illustrated below:

7	6	5	4	3	2	1	0
Reserved	Frame alignment word						

The synchronization pattern transmitted by the AU to the ISTE is asymmetric to the pattern transmitted by the ISTE to the AU so that the data stream will avoid misalignment caused by crosstalk noise. The FAW transmitted by the AU to the ISTE shall be:

6	5	4	3	2	1	0
0	1	0	0	1	1	1

where the least significant bit of the FAW (bit 0) is transmitted first. The FAW transmitted by the ISTE to the AU shall be:

6	5	4	3	2	1	0
1	0	1	1	0	0	0

where the least significant bit of the FAW (bit 0) is transmitted first.

Bit 7 is reserved for future standardization use. It shall be transmitted as a logic "0," and ignored upon receipt by both the ISTE and the AU.

9.3.1.1 Synchronization procedure

Except for the encoding of the synchronization pattern, the synchronization procedures of the ISTE and the AU are equivalent.

While in the disabled state, the PS sublayer's transmitter shall not operate. When the PS transmitter is activated, the transmit frame shall continuously generate the appropriate synchronization pattern in the FAW field of each TDM frame.

While in the powered down or disabled state, the PS sublayer's receive framer shall not operate. When initialized, the receive framer enters the unsynchronized state, in which it shall search for the appropriate FAW pattern in the received bit stream. Upon correctly receiving three consecutive FAW patterns at TDM frame length intervals, the receive framer shall enter the synchronization or LOCK state. The multiplexed data stream, except for the SYN field, is scrambled to reduce the probability of false synchronization.

While in the LOCK state, the receive framer shall compare each received synchronization pattern with the expected value. If three consecutive synchronization patterns are received in error, the receive framer shall return to the unsynchronized state.

The current state of the PS sublayer's receive framer shall be reported to the far end in each transmitted TDM maintenance field, as part of the local receiver frame status bit.

9.3.2 Scrambling

Scrambling distributes the transmitted spectrum, thus reducing EMI and facilitating receiver functions such as clock recovery and adaptive equalization.

A self-synchronizing (or feed-through) type scrambler shall be employed at a transmitting side and a corresponding descrambler shall be provided at a receiving end. The scrambling shall not apply to the SYN field of the TDM frame. The generator polynomial shall be $1 + x^{-18} + x^{-23}$ and the corresponding scrambler/descrambler configurations are as shown in figure 9-16.

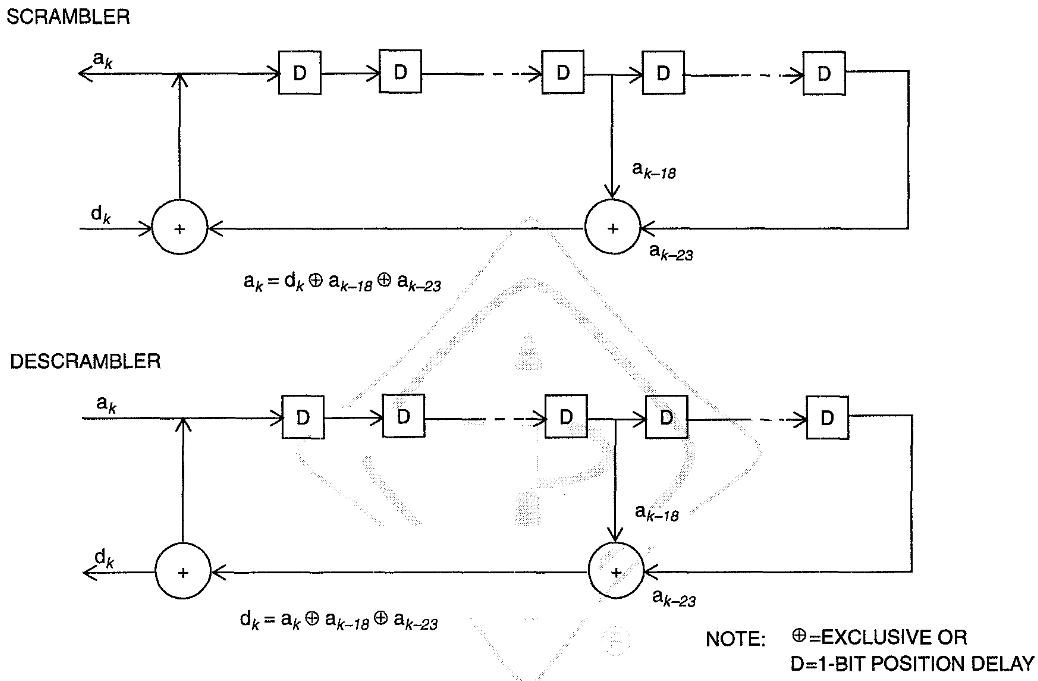


Figure 9-16—Self-synchronizing type scrambler and descrambler

9.3.3 Parity bits

Two parity bits shall be provided for monitoring the bit errors at the PHY Layer. These bits are located in the lower two bits of the TDM_MTN field.

9.3.3.1 Odd parity bit

These bits are separate from the functionality expressed by the remainder of TDM_MTN field. The odd parity bit, PODD, occupies the bit 1 position of the TDM_MTN field. This bit is transmitted, in TDM frame "n," as the modulo two sum of the contents of each odd-numbered bit position in TDM frame "n-1" (see 9.3.4).

9.3.3.2 Even parity bit

The even parity bit, PEVN, occupies the bit 0 position of the TDM_MTN field. This bit is transmitted, in TDM frame "n," as the modulo two sum of the contents of each even-numbered bit position in TDM frame "n-1" (see 9.3.4).

9.3.4 Parity generation and checking procedures

Transmission errors are estimated in the PS sublayer by generating parity in transmitted TDM frames, and by checking parity in received frames. Detected errors are reported locally through the management interface to layer management, and remotely through the TDM maintenance field (TDM_MTN).

The parity generator shall operate only while the PS sublayer's transmitter is in the enabled state.

The parity checker shall operate only while the PS sublayer's receive framer is in the LOCK state.

Parity shall be calculated over the entire TDM frame, including the contents of the SYN field. Parity shall be calculated independently for the odd and even numbered bits within each TDM frame. For purposes of parity calculations, the "zero," "2nd," "4th," etc., bit positions of the TDM frame, in time order of transmission, are even-numbered bits. The contents of the odd bit parity position of TDM frame "n" shall be transmitted as the sum, modulo two, of the contents of each odd-numbered bit position in TDM frame "n-1," starting with the second transmitted bit position of the SYN field, and ending with the last transmitted bit of frame "n-1."

Similarly, the contents of the even bit parity position of the TDM frame "n" shall be transmitted as the sum, modulo two, of the contents of each even-numbered bit position of the TDM frame "n-1," starting with the first transmitted bit position of the SYN field, and ending with the second to last transmitted bit of frame "n-1."

The states of the odd and even bit parity positions in the first transmitted TDM frame shall be set to "0."

The receiver shall independently calculate the sums, modulo two, of the contents of each odd and even numbered bit position in each frame "n" and compare the calculated results with the values received in the odd and even bit parity positions of each frame "n+1."

If either of the results calculated for frame "n" differs from the value received in frame "n+1," the next transmitted parity error bit in the TDM maintenance field shall be set to logic "1." Otherwise, it will be set to logic "0."

A local parity error counter shall be incremented for each received TDM frame "n+1" in which the contents of either or both of the odd or even bit parity positions differs from the results calculated for frame "n."

A remote parity error counter shall be incremented for each received TDM frame in which the contents of the parity error bit in the TDM_MTN field is set to logic "1."

9.3.5 Maintenance

In reference to the PS sublayer's transmit and receive functional diagrams (see figures 9-14 and 9-15), the TDM_MTN field is provided by the management entity and the PS sublayer transmitter circuit multiplexes this into the bit stream TX PS_DATA and sends this combined stream down to the PMD sublayer as the bit stream TX PMD_DATA.

In the receiver, the TDM_MTN field is demultiplexed from the incoming bit stream RX PMD_DATA and is presented to management.

The second octet of each TDM frame, the TDM maintenance (TDM_MTN) field, is used to transmit local PHY Layer status and control information to the far end of the link. The field is encoded as follows:

7	6	5	4	3	2	1	0
RSRV	RLBRESP	RLBREQ	LLB	LOCK	PERR	PODD	PEVN

The summarized list of functions is given in table 9-8.

Table 9-8—Functions of TDM maintenance fields

Bit	Name	Function	Value
7	RSRV	Reserved	Transmitted as 0 and ignored upon receipt
6	RLBRESP	Remote loopback response	0 = Deactivated 1 = Activated
5	RLBREQ	Remote loopback request	0 = Deactivate request 1 = Activate request
4	LLB	Local loopback status	0 = Deactivated 1 = Activated
3	LOCK	Local receiver frame status	0 = Unlocked 1 = Locked
2	PERR	Local parity check status	0 = Passed 1 = Failed
1	PODD	Cumulative odd bits parity checksum	(x)
0	PEVN	Cumulative even bits parity checksum	(x)

The functionality of these bits of the TDM_MTN field is described in the following subclauses.

9.3.5.1 Remote loopback response

This bit is transmitted as a logic "1," in response to the reception of a remote loopback request, to indicate that the remote loopback path is active.

A station receiving this bit as a logic "1" can assume that the contents of a received TDM frame, exclusive of the SYN and TDM_MTN fields, are being retransmitted by the remote station as received.

9.3.5.2 Remote loopback request

This bit is transmitted as a logic "1" to request that the remote station activate or maintain a remote loopback function.

A station receiving this bit as a logic "1" should respond at the earliest opportunity by transmitting the remote loopback response bit as a logic "1," and by retransmitting the contents of the TDM frame, exclusive of the SYN and TDM_MTN fields, as received. The loopback path shall include the scrambler and descrambler, and exclude the parity check and generation circuit functions. As a recommendation, this loopback should be as close to the PS-to-HMUX service boundary as possible.

A station receiving this bit as a logic "0" should transmit the remote loopback response bit as a logic "0" and disable its remote loopback circuit.

9.3.5.3 Local loopback status

This bit is transmitted as a logic "1" while any local loopback function is active. It is transmitted as a logic "0" otherwise. A station receiving this bit as a logic "1" can assume that the sending station is not delivering incoming TDM frames across its PMD to the IEEE 802.9 line interface.

When the local LME decides to invoke a local loopback function, it will send the signal LOCAL_LOOP_ON to the PMD sublayer transmitter (see figure 9-14). This will cause the PMD_DATA stream to be routed back to the PMD sublayer receiver (see figure 9-15); it will not be transmitted onto the physical medium. It is recommended that this loopback be exercised at as close a point to the physical medium attachment point as possible. The transmitter function will transmit the SYN and TDM_MTN fields as part of the information stream delivered. Note that the placement of the physical loopback module is an implementation choice.

9.3.5.4 Local receiver frame status

This bit is transmitted as a logic "0" while the receive framer is in the disabled or unsynchronized states, and a logic "1" while the receive framer is in the synchronized state.

A station receiving this bit as value "0" can assume that the sending station is unable to acquire or reacquire frame synchronization, and therefore cannot process the contents of incoming TDM frames, including the TDM maintenance field.

9.3.5.5 Local parity check bit result

This bit is transmitted as a logic "1" if the result of the parity check on the previously received TDM frame failed. It is transmitted as a logic "0" if the result of the parity check on the previously received TDM frame passed.

If this bit is received as a logic "1," a transmitted parity error counter is incremented by one. This counter is an MO, and supports the operations of GET and SET (to default), which return the current value of the counter and reset it to logic "0," respectively.

9.3.5.6 Cumulative odd parity bit

The cumulative odd parity bit, PODD, occupies the bit 1 position of the TDM_MTN field. This bit is transmitted, in TDM frame "n," as the modulo two sum of the contents of each odd-numbered bit position in TDM frame "n-1."

9.3.5.7 Cumulative even parity bit

The cumulative even parity bit, PEVN, occupies the bit 0 position of the TDM_MTN field. This bit is transmitted, in TDM frame "n," as the modulo two sum of the contents of each even-numbered bit position in TDM frame "n-1."

9.3.5.8 Hysteresis

Each of the four TDM_MTN field bit positions, LOCK, LLB, RLBREQ, and RLBRESP, is protected from transmission errors by hysteresis.

In the receiver, each of these four bit positions is associated with an M-stage counter and a Boolean flag variable. Each counter is initially set equal to value M/2, and each flag variable is set to False. It is recommended that M/2 be set to value "3."

For each LOCK, LLB, RLBREQ, or RLBRESP bit received as a logic "1," the associated counter, if not equal to "M-1," is incremented by one. When the counter becomes equal to "M-1," the associated flag variable, if False, becomes True (or if True, remains True).

For each bit received as a zero, the associated counter, if not equal to zero, is decremented by one. When the counter becomes equal to zero, the associated flag variable, if True, becomes set to False (or if False, remains False).

If a counter is incremented to a value less than "M-1," or decremented to a value greater than "0," the associated variable retains its current value.

Figure 9-17 illustrates the up/down counter and the flag register.

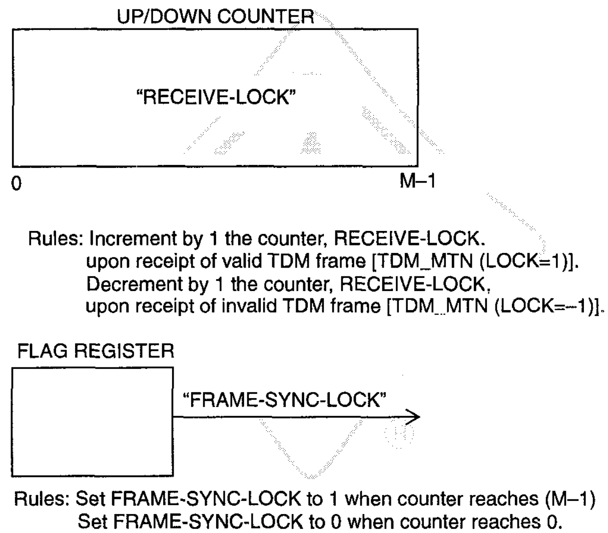


Figure 9-17—Model of counter operation

9.3.6 RESERVED field

The RESERVED field occupies the third octet in the TDM frame. This field is provided by the management entity and the PS sublayer transmitter circuit multiplexes this into the bit stream TX PS_DATA and sends this combined stream down to the PMD sublayer as the bit stream TX PMD_DATA.

In the receiver, the reserved field is demultiplexed from the incoming bit stream RX PMD_DATA and is presented to management and passed up to a higher layer.

9.3.7 PS sublayer initialization procedures

9.3.7.1 General

This subclause describes the initialization procedure for the PHY Layer. The frame synchronization field is described in 9.3.1. In this model, the PHY Layer is in the powered-on condition. Therefore, the following procedure specifies the initialization procedure after power-up, and the error recovery procedure for the PHY Layer connection. The PS sublayer executes a handshake procedure with the peer entity's PS sublayer as described in the FSM in the following subclauses.

9.3.7.2 Definition of external signals, internal signals, and state variables

The following is a glossary of terms used to describe internal control registers and signals employed in the execution of this initialization procedure. Positive logic is used as a convention; therefore, "1" and "TRUE" are equivalent.

9.3.7.2.1 External signals

The following primitive signals appear as external stimulus to the PHY initialization FSM:

LM_PH_ACTION(activate) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are activated and the PHY Layer initialization state machine will enter the AWAITING_LOCAL_LOCK state.

LM_PH_ACTION(deactivate) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are deactivated and the PHY Layer initialization state machine will enter the INACTIVE state.

LM_PH_ACTION(reset) — When this management request is issued, the PMD, the PS, and the HMUX sublayers are reinitialized and the PHY Layer initialization state machine will enter the AWAITING_LOCAL_LOCK state.

TDM_MTN(LOCK=0) — This indicates that the PS sublayer has received a TDM frame with the LOCK indication bit of the TDM maintenance field set to logic "0" (i.e., unlocked).

TDM_MTN(LOCK=1) — This indicates that the PS sublayer has received a TDM frame with the LOCK indication bit of the TDM maintenance field set to logic "1" (i.e., locked).

FRAME_SYNC(0) — This indicates that the PS sublayer has lost frame synchronization according to the criteria of the frame protection defined in 9.3.1.1.

FRAME_SYNC(1) — This indicates that the PS sublayer has established the frame synchronization following the procedure defined in 9.3.1.1.

9.3.7.2.2 Internal signals

These are the internal signals that are generated by the functional processes within the PHY initialization FSM.

9.3.7.2.3 State names

These are commonly referred to as "flags" and "event/error counters." The following are definitions of the states used in the functional model for the PHY initialization FSM:

INACTIVE — The INACTIVE state is the deactivated condition. The PHY initialization FSM will vector into the AWAITING_LOCAL_LOCK state when it receives the LM-PH-ACTION(activate) signal (i.e., power-up). In this state, the FSM will not respond to any stimulus except for messages from its LME.

AWAITING_LOCAL_LOCK	The AWAITING_LOCAL_LOCK state is a state in which the PS sublayer is in the loss of frame synchronization condition. In this state, the ISTE can send the TDM frame using its local clock. But once the ISTE has successfully extracted the clock for the received bit stream sent from the AU, it must use this clock to send the TDM frame. The PHY initialization FSM will vector into the AWAITING_REMOTE_LOCK state when it detects frame synchronization.
AWAITING_REMOTE_LOCK	The AWAITING_REMOTE_LOCK is a state in which the PS sublayer is receiving a TDM frame with the LOCK bit set to logic "0" from the peer PS sublayer entity. The PHY initialization FSM will vector into the ESTABLISHED state when it receives TDM_MTN(LOCK=1) a certain number of times (to satisfy the hysteresis counter RECEIVE_LOCK). If the FSM detects a loss of frame synchronization, then it will vector into the AWAITING_LOCAL_LOCK state.
ESTABLISHED	The ESTABLISHED state is the state in which the PS sublayer FSM is in the frame synchronization condition and is receiving the TDM_MTN(LOCK=1) from the peer entity's PS sublayer. The PHY initialization FSM will vector back into the AWAITING_LOCAL_LOCK state when it detects a loss of frame synchronization. The FSM will vector back into the AWAITING_REMOTE_LOCK state when it receives TDM_MTN(LOCK=0) a certain number of times (to satisfy the hysteresis counter RECEIVE_LOCK).

9.3.7.2.4 Managed objects

The following MO within the FSM is controlled or "managed" by the management entity within the ISTE and AU:

RECEIVE_LOCK	This managed 8-bit up/down counter is used as a hysteresis mechanism to monitor the LOCK signal from the remote peer. It is incremented upon receipt of a TDM_MTN(LOCK=1) signal. It is decremented upon receipt of a TDM_MTN(LOCK=0) signal.

9.3.8 FSM for PS sublayer initialization

Figure 9-18 and tables 9-9a and 9-9b describe the FSM for the PHY initialization procedure.

9.3.9 PHY Layer maintenance

9.3.9.1 Overview

The PHY Layer maintenance consists of a local loopback procedure and a remote loopback procedure. These loopback procedures are described in the following subclauses. The PHY Layer maintenance request/ status functionality is transported via the TDM_MTN field which is described in 9.3.

9.3.9.2 Loopback

The TDM_MTN field (see 9.3.5) contains maintenance status for the purpose of exchanging local loopback status and to exchange remote loopback invocation request/response information with the peer PHY entity.

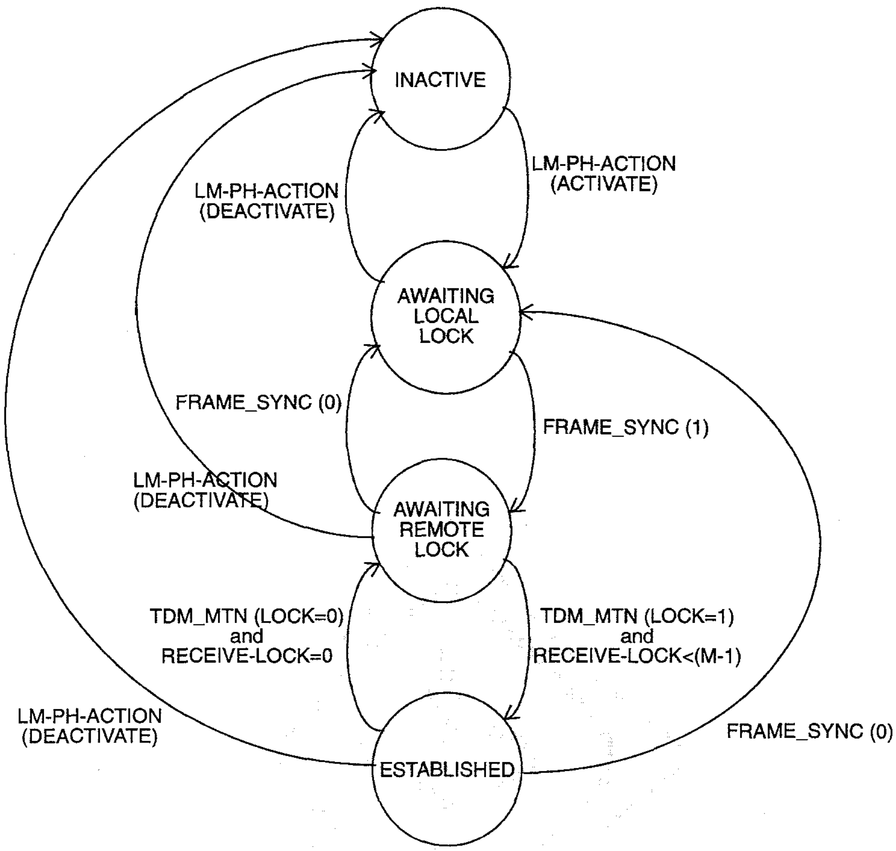


Figure 9-18—FSM state diagram for PS sublayer initialization

9.3.9.2.1 Local loopback procedure

In a maintenance inquiry, it is possible for either the AU or the ISTE to invoke a local loopback operation. Figure 9-19 illustrates this loopback operation. In the AU, a network management task could initiate a local loopback test. The AU would respond by providing a loopback of the TDM information stream in the PMD sublayer. The location of the loopback function is an implementation choice. The sequence of steps used in the management invocation of the loopback is also a function of the implementation approach. The AU provides the clock timing to support this loopback. The only portion of the aggregate TDM stream that is sent or received from the physical medium during this test is the SYN field and the TDM_MTN field. The PS sublayer, within the AU, shall set the logical loopback status (LLB) bit to a logic "1" in the TDM_MTN field to notify the ISTE of the maintenance outage.

While honoring a local loopback request, the endpoint (ISTE or AU) shall operate as follows:

a) Transmit framer operation is normal; i.e., the received SYN field is not looped back. This prevents the requesting endpoint from having to reconfigure its receive framer to search for the complementary synchronization pattern.

b) Transmit TDM maintenance field generation is normal; i.e., the received TDM maintenance field is not looped back, and the transmitted TDM maintenance field reflects current local conditions. This provides the requesting endpoint with continuous monitoring of parity errors in each direction, and the remote synchronizer state during remote loopback.

Table 9-9a—FSM state table for PS sublayer initialization for INACTIVE, AWAITING_LOCAL_LOCK, and AWAITING_REMOTE_LOCK states

Current	Event	Action(s)	Next state
INACTIVE	LM-PH-ACTION(activate)	SET_VARIABLES_ AND TIMERS SEND_TDM_MTN (LOCK=0)	AWAITING_ LOCAL_LOCK
	LM-PH-ACTION	NO_ACTION	INACTIVE
	LM_PH_ACTION(reset)	SET_VARIABLES_AND_ TIMERS_TO_DEFAULT	INACTIVE
AWAITING_ LOCAL_LOCK	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AWAITING_ LOCAL_LOCK
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (RESET)	AWAITING_ LOCAL_LOCK
	FRAME_SYNC (0)	SEND_TDM_MTN (LOCK=0)	AWAITING_ LOCAL_LOCK
	FRAME_SYNC (1)	SEND_TDM_MTN (LOCK=1) RECEIVE-LOCK:=(M/2)	AWAITING_ REMOTE_LOCK
AWAITING_ REMOTE_LOCK	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	AWAITING_ REMOTE_LOCK
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	SET_VARIABLE_AND_ TIMERS_TO_DEFAULT	AWAITING_ REMOTE_LOCK
	FRAME_SYNC (0)	SEND_TDM_MTN (LOCK=0)	AWAITING_ LOCAL_LOCK
	FRAME_SYNC (1)	NO_ACTION	AWAITING_ REMOTE_LOCK

Optionally, a maintenance application within the ISTE may invoke the local loopback procedure. The ISTE will effect a loopback of the aggregate TDM stream within its PMD sublayer. The location of this loopback function is an implementation choice. The ISTE shall provide a local clock timing source for this loopback operation. No portion of the aggregate information stream is sent or received from the physical cable medium. The PS sublayer within the ISTE shall set the LLB bit to a logic "1" in the TDM_MTN field to notify the AU of the maintenance condition.

9.3.9.2.2 Remote loopback

The TDM_MTN field supports the exchange of a remote loopback request bit to request that the remote station activate or maintain a loopback function. Associated with this request is a response bit to allow the remote station to affirm this maintenance request. This protocol is symmetrical in that either the ISTE or the AU may invoke this request. The invocation of the request is optional. However, the support of the response is mandatory in the sense that a response (either "1" or "0") is required. For example, upon receipt of an activated (logic "1") remote loopback request, the receiving PHY entity must respond with an appropriate response on the RLBRESP bit of the TDM_MTN field.

Table 9-9b—FSM state table for PS sublayer initialization for AWAITING_REMOTE_LOCK and ESTABLISHED states

Current state	Event	Action(s)	Next state
AWAITING_ REMOTE_LOCK	TDM_MTN (LOCK=0)	DECREMENT_RECEIVE LOCK	AWAITING_ REMOTE_LOCK
	TDM_MTN (LOCK=1) and RECEIVE-LOCK<(M-1)	INCREMENT_RECEIVE-LOCK	AWAITING_ REMOTE_LOCK
	TDM_MTN (LOCK=0) and RECEIVE-LOCK=(M-1)	INCREMENT_RECEIVE-LOCK LM-PH-NOTIFICATION (ESTABLISHED)	ESTABLISHED
ESTABLISHED	LM-PH-ACTION(activate)	LM-PH-NOTIFICATION (INVALID_COMMAND)	ESTABLISHED
	LM-PH-ACTION(deactivate)	NO_ACTION	INACTIVE
	LM-PH-ACTION(reset)	LM-PH-NOTIFICATION (INVALID_COMMAND)	ESTABLISHED
	FRAME_SYNC (0)	SEND_TDM_MTN (LOCK=0) LM-PH-NOTIFICATION (released)	AWAITING_ LOCAL_LOCK
	FRAME_SYNC (1)	NO_ACTION	ESTABLISHED
	TDM_MTN (LOCK=1) and RECEIVE_LOCK<(M-1)	INCREMENT_RECEIVE-LOCK	ESTABLISHED
	TDM_MTN (LOCK=0) and RECEIVE_LOCK>(0)	DECREMENT_RECEIVE-LOCK	ESTABLISHED
	TDM_MTN (LOCK=0) and RECEIVE-LOCK=(0)	LM-PH-NOTIFICATION (released)	AWAITING_ REMOTE_LOCK

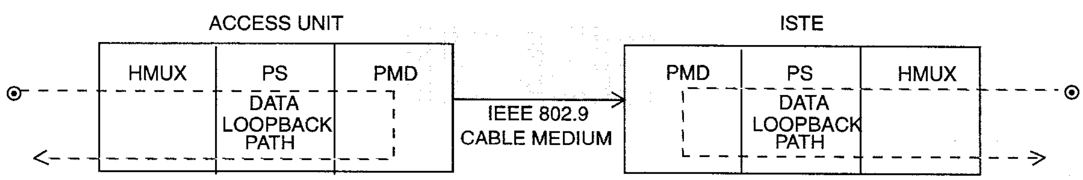


Figure 9-19—Data loopback path under local loopback

Figure 9-20 illustrates the optional situations in which

— The AU has initiated the remote loopback procedure.
— The ISTE has initiated the loopback procedure.

For the event in which a network management process, associated with the AU, initiates the remote loopback operation, it will set the RLBREQ bit to a logic "1" in the TDM_MTN field. Upon receipt of this command, the ISTE's PS sublayer will effect a loopback of the data stream at a point close to the

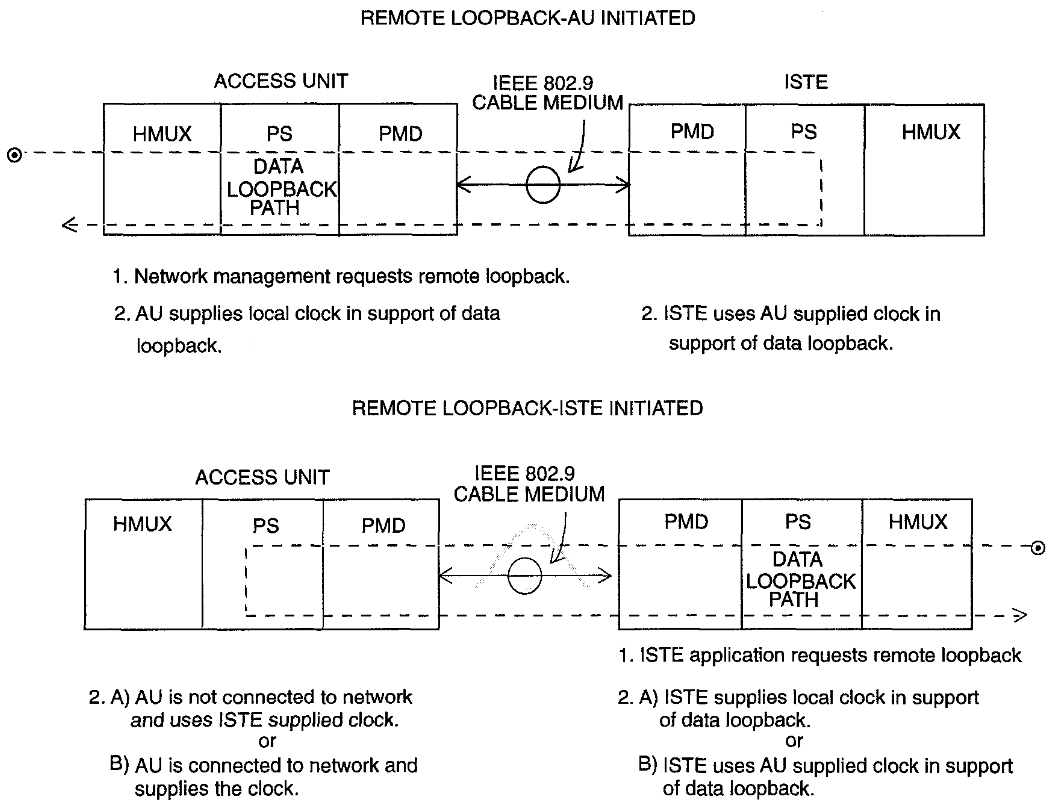


Figure 9-20—Data loopback path under remote loopback

HMUX-to-PS sublayer service interface. The location of this loopback function is an implementation choice. The ISTE shall use the extracted clock from the received data stream as the source of the transmitted data clock. Upon the enabling of this loopback operation, the ISTE shall set the RLBRESP bit to a logic “1” in the transmitted TDM_MTN field to acknowledge this maintenance request. A second usage of the remote loopback would be the ISTE initiated procedure. In this case, the ISTE would set the RLBREQ bit to a logic “1.” Upon receipt of this command, the AU will effect a loopback of the data stream at a point close to the HMUX-to-PS service boundary. The location of this loopback function is an implementation choice. There are two options for the ISTE to use as the transmitter timing source:

- For bringing up a portion of a network from a “cold” starting point, the ISTE may choose to supply the clock.
- If the AU is connected to the network clock hierarchy, the ISTE shall use as its transmitter clock that timing which is derived from the received data stream (AU clock).

Upon the enabling of this loopback operation, the ISTE shall set the RLBRESP bit to a logic “1” in the transmitted TDM_MTN field to acknowledge this maintenance request. If the implementation in either the AU or the ISTE does not support the remote loopback procedure, it has the requirement that it respond with the RLBRESP bit set to a logic “0.”

In the event that the AU and the ISTE simultaneously request the remote loopback function (by transmitting RLBREQ = 1), the AU shall have the priority. That is, the ISTE shall respond to the AU request by activating the remote loopback path and transmitting RLBRESP = 1. The ISTE may continue to transmit RLBREQ = 1 in this condition, but should not consider the AU’s failure to respond with RLBRESP = 1 as an error condition.

While honoring a remote loopback request, the endpoint (ISTE or AU) shall operate as follows:

- Receive framer operation is normal to maintain the state of the local synchronization variable.
- Receive descrambling operation is normal to allow processing of the TDM maintenance field.
- Received TDM maintenance field checking is normal; i.e., received parity errors continue to be detected and counted, remotely reported parity errors continue to be counted, and the state of the remote receive framer and loopback variables are maintained.
- Transmit framer operation is normal; i.e., the received SYN field is not looped back. This prevents the requesting endpoint from having to reconfigure its receive framer to search for the complementary synchronization pattern.
- Transmit scrambling operation is normal.
- Transmit TDM maintenance field generation is normal; i.e., the received TDM maintenance field is not looped back, and the transmitted TDM maintenance field reflects current local conditions. This provides the requesting endpoint with continuous monitoring of parity errors in each direction, and the remote synchronizer state during remote loopback.
- The remainder of each received descrambler TDM frame is looped back to the scrambled, rescrambled, and retransmitted without modification. This provides the requesting endpoint with an opportunity to inject arbitrary patterns into arbitrary channels (except for the SYN and TDM maintenance fields), for example to perform bit error rate (BER) testing.
- If local frame synchronization is lost while honoring the remote loopback request, an endpoint shall continue to honor the request until synchronization is reacquired and the presence or absence of the request can be redetermined. When local frame synchronization is lost, the data being looped-back may be indeterminate.

9.4 PMD sublayer for 4.096 Mb/s application

Figure 9-21 illustrates the detailed functional model for the transmitter circuit for the 4.096 Mb/s PMD sublayer. Figure 9-22 illustrates the detailed functional model for the receiver circuit for the 4.096 Mb/s PMD sublayer.

9.4.1 Bit rate

The bit rate at the PMD sublayer/PS sublayer service boundary shall be 4.096 Mb/s. This assumes two unshielded twisted-pair wires between the ISTE [or the integrated services terminal adapter (ISTA)] and the AU. The 4.096 Mb/s includes all the required overhead fields such as those required for frame synchronization, maintenance, and parity bits.

The clock rate of the AU shall be within ±32 ppm and the clock rate of the ISTE without receiving any signal from the AU shall be within ±100 ppm.

9.4.2 Line code

The line code for the ISLAN interface between the PMD sublayer and the physical medium of 4.096 Mb/s shall be Binary Partial Response Class IV.

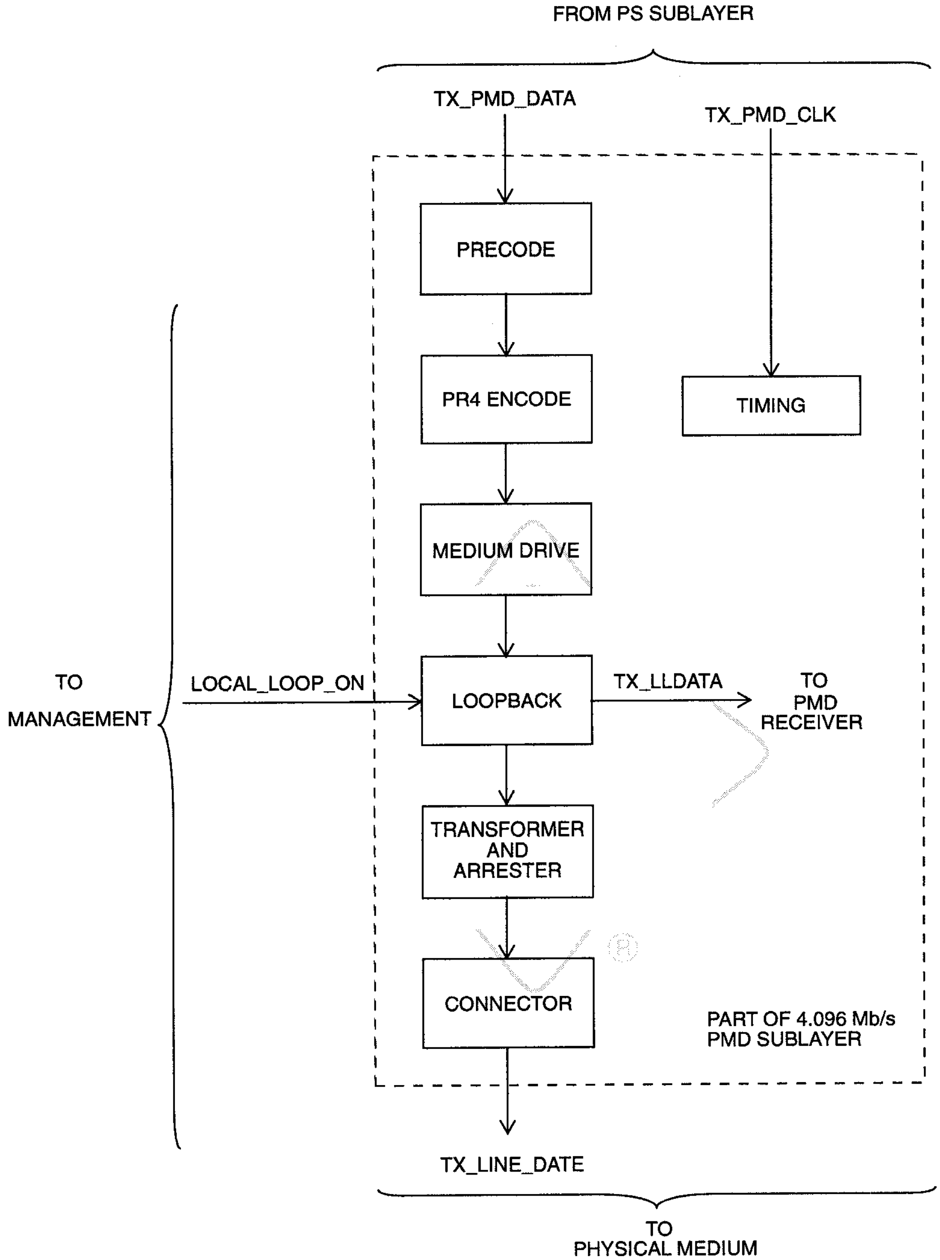


Figure 9-21—Model of the transmitter for the 4.096 Mb/s PMD sublayer

9.4.2.1 Precoding

The incoming bit stream is first precoded so that the decision at the receiving end may be made regardless of the previous decision result, thus avoiding the error propagation effect. Let the incoming bit sequence (referred to as "the input data to the transmitter" hereafter) be denoted as a_k, where the subscript k indicates discrete time measured in T, the signalling interval. The precoding operation can be expressed as converting the input sequence, a_k, to the output sequence, b_k, by the following equation:

$b_k = a_k \text{ ExOr } b_{k-2}$

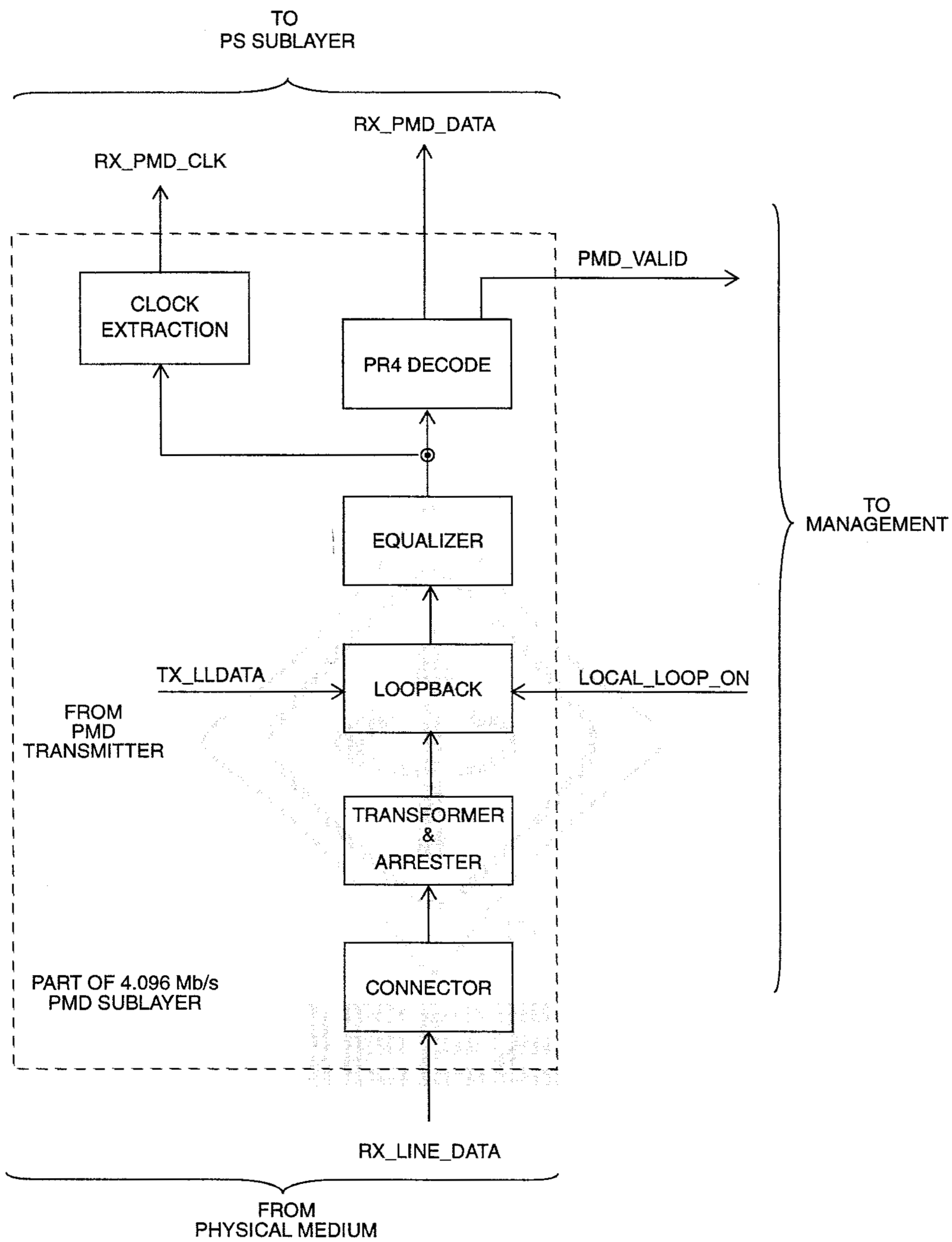


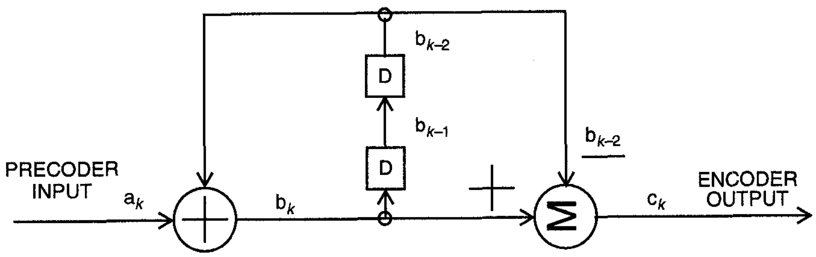
Figure 9-22—Model of the receiver for the 4.096 Mb/s PMD sublayer

The structure of the precoder is shown in figure 9-23. The output sequence of the precoder is still binary.

9.4.2.2 Encoding

The encoding is operated on the precoded sequence such that the encoded sequence, c_k, has the following relation with the recoded sequence, b_k:

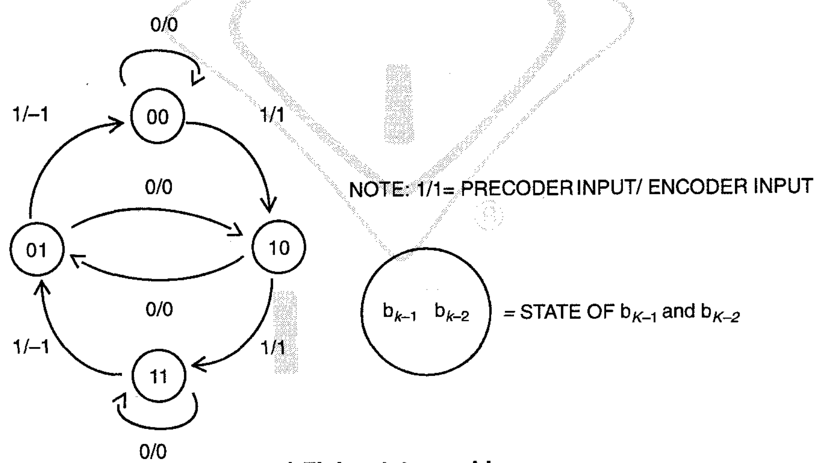
$$c_k = b_k - b_{k-2}.$$



a) Block diagram

a_k	b_{k-1}	b_{k-2}	b_k	c_k
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	1	0
1	0	0	1	1
1	0	1	0	–1
1	1	0	1	1
1	1	1	0	–1

b) Truth value table



c) Finite state machine

Figure 9-23—Precoder and encoder

The encoding shall be done at the transmitting end. The structure of the encoder is shown in figure 9-23 together with the precoder.

9.4.2.3 Decoding

The decoding for the precoded Partial Response Class IV is done by a combination of sample/slice (three level) and modulo 2 operation. A typical decoder may be implemented by a full wave rectifier followed by a comparator. Provisioning of maximum likelihood sequence decoding or other techniques are outside of the scope of this standard.

9.4.3 Transmitter characteristics

The transmitter output signal is measured at the connector of the transmitter to medium cable. All of the reference models of medium cable, which are defined in this subclause, shall be used as loads to the transmitter. Unless otherwise indicated, the differential output signal and the common mode output signal shall be measured respectively using the setup shown in figures 9-24 and 9-25.

The reference model circuits of medium cables are constructed as shown in figures 9-26 through 9-28. Those circuits are the models of a few medium cables whose lengths are different from each other. The circuit illustrated in figure 9-26 is the model of a 150 m length of 24 gauge D-type inside wire (DIW). Figure 9-27 shows the model of a 300 m length of 24 gauge DIW, and figure 9-28 shows the model of a 450 m length of 24 gauge DIW.

Unless otherwise indicated, the value of resistance is in ohms (Ω), the value of inductance is in microhenries (μH), and the value of capacitance is in farads (F). The direction of signal flow is from left to right.

9.4.3.1 Transmitter output signal pulse shape

The isolated pulse of the differential output signal shall be defined as an output of the transmit filter. The input of the filter is a 50% duty return-to-zero pulse whose rise and fall times are within 20 ns in length. The transmit filter is a second-order Butterworth low-pass filter whose cutoff frequency is 2 MHz. The errors of the cutoff frequency and the Q-value should be less than 10%. The frequency characteristics of the transmit filter are shown in figure 9-29 and table 9-10.

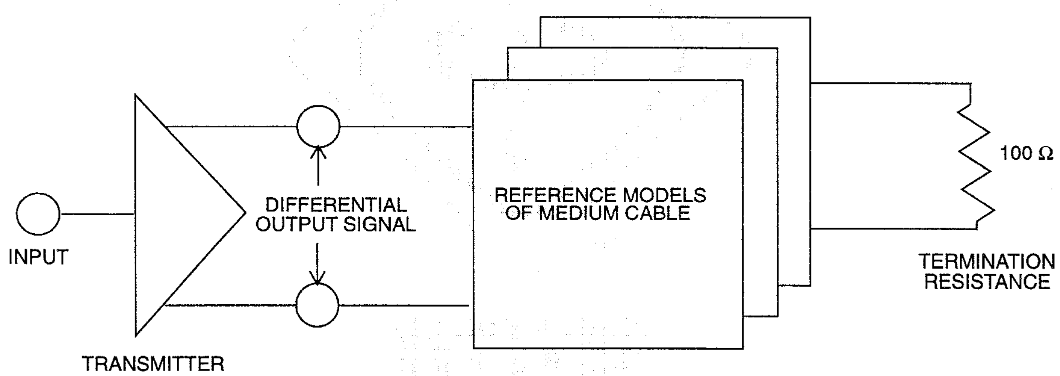


Figure 9-24—Measurement set of differential output signal

The typical pulse shape, as observed at the output of the transmit filter, is illustrated in figure 9-30 for the case where the input data series into the precoder is '...10001000...'.

In figure 9-30, the critical points are measured in units of time (T) and amplitude (A).

9.4.3.2 Transmitter amplitude

The amplitude of the differential output signal is 3 $V_{p\text{-}p}$.

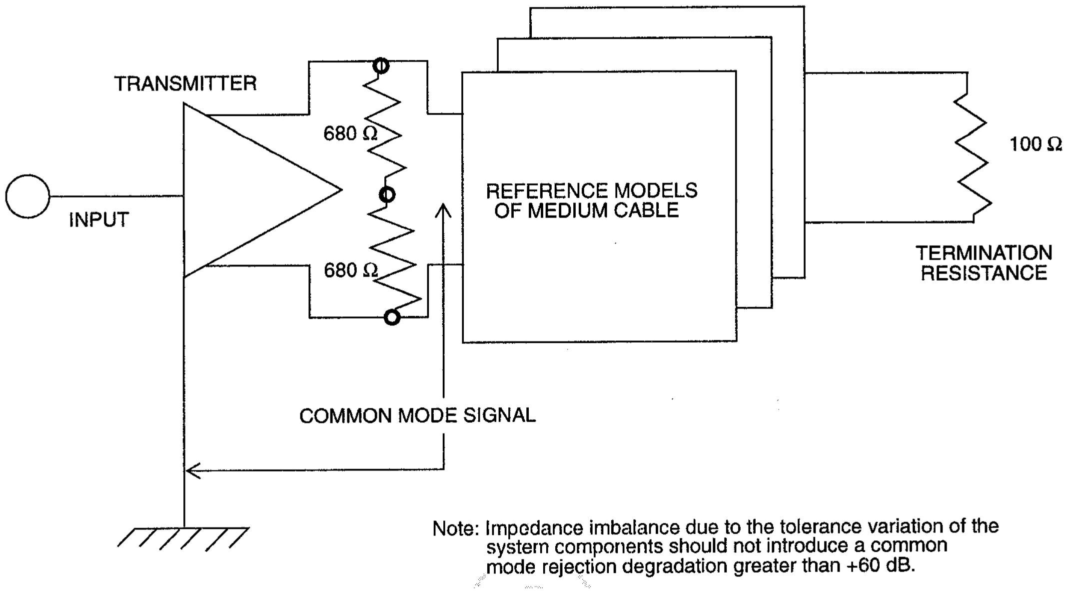


Figure 9-25—Measurement set of common mode output signal

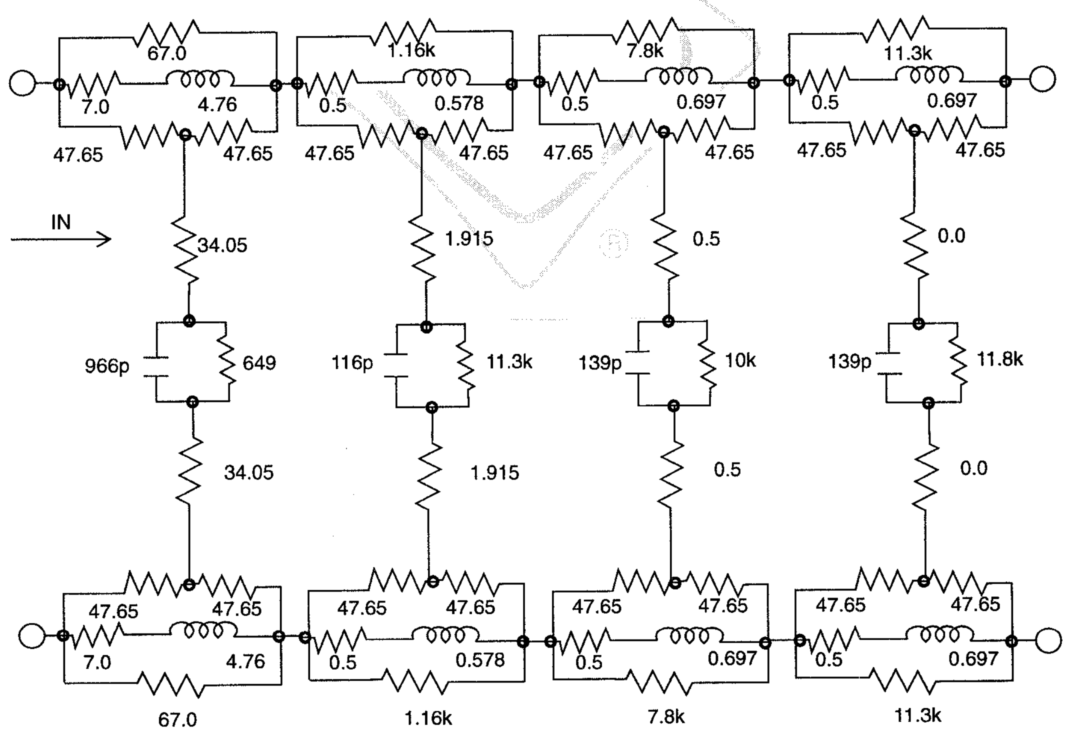


Figure 9-26—Reference model of 150 m medium cable length

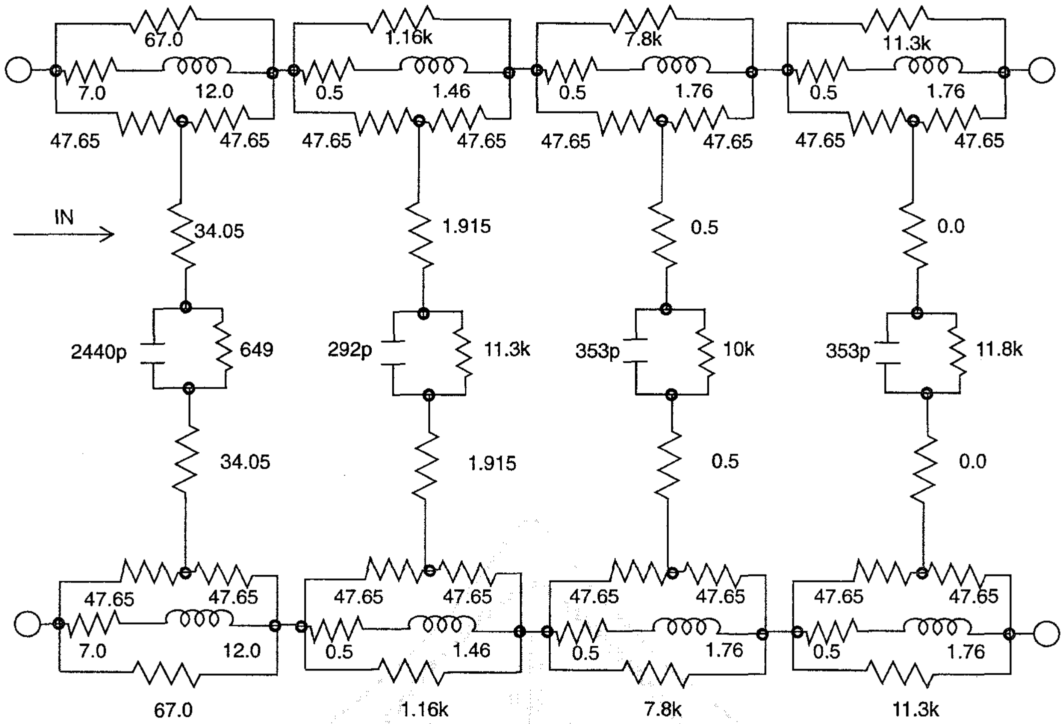


Figure 9-27—Reference model of 300 m medium cable length

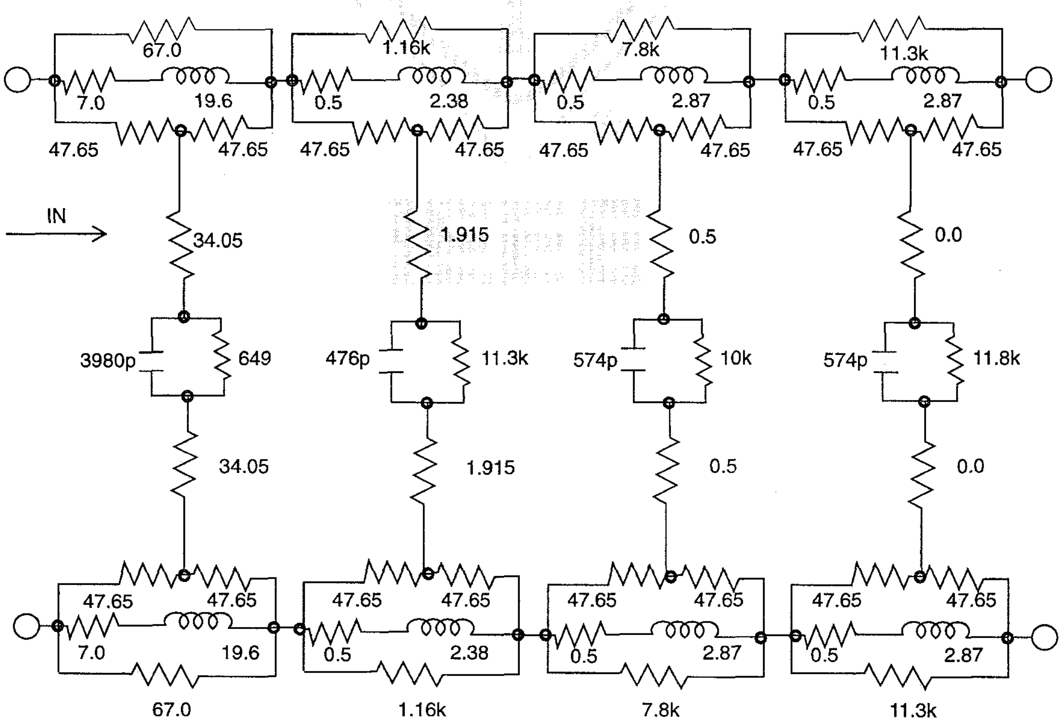


Figure 9-28—Reference model of 450 m medium cable length

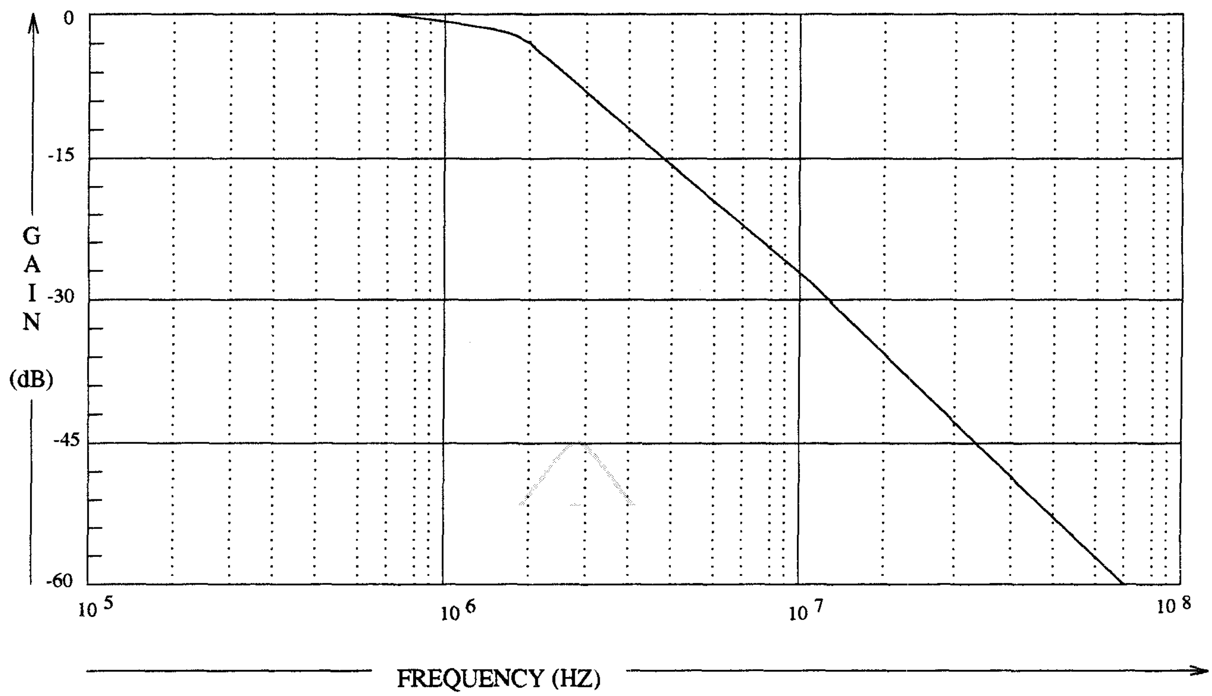


Figure 9-29—Frequency characteristics of transmit filter for 4.096 Mb/s

9.4.3.3 Transmitter output impedance

The differential output impedance of the transmitter, Z_T, shall be such that the reflection attenuation, defined as

$$20\log_{10}\left\{\frac{|Z_T + 100|}{|Z_T - 100|}\right\}$$

exceeds 15 dB at 1.024 MHz with 100 Ω termination.

The common mode to differential mode impedance balance of the transmitter is defined as

$$20\log_{10}\left\{\frac{E_{cm}}{E_{dif}}\right\}$$

where E_{cm} is an externally applied ac voltage as shown in figure 9-31.

The impedance balance shall exceed 44 dB at 1.024 MHz.

9.4.3.4 Transmitter output power spectrum density

The power spectrum of the differential output signal for a pseudo-random sequence shall be below the upper trajectory of figure 9-32. The lower curve of this figure shows the result of calculation under the condition of the typical pulse shape shown in figure 9-30. The power spectrum of common mode current for a pseudo-random sequence shall be below the upper trajectory of figure 9-33. The lower curve is the result of calculation under a typical condition. The pseudo-random sequence is to be generated by applying binary 1's to the IEEE 802.9 scrambler specified in 9.3.2.

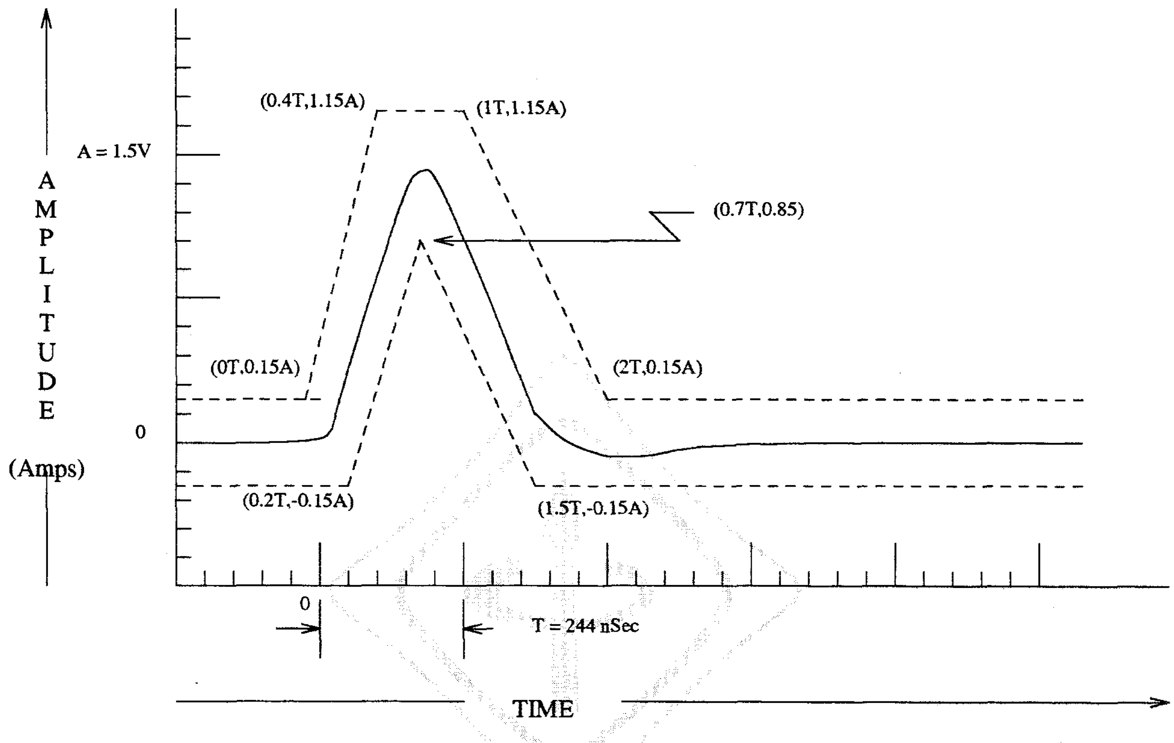


Figure 9-30—Pulse shape of 4.096 Mb/s transmitter output

9.4.3.5 Transmitter output jitter

The transmitted signal zero-crossings shall deviate from the idealized zero-crossings by no more than ±10 ns. The input data series into the precoder is "...1111...."

9.4.3.6 Transmitter common mode requirement

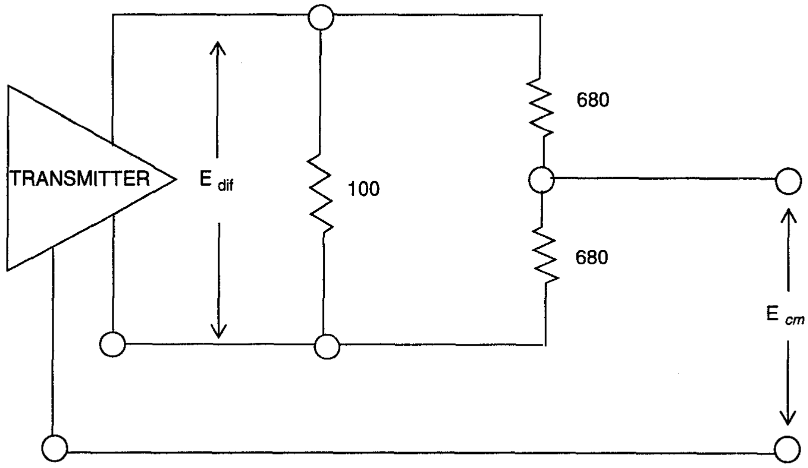
The transmitter shall meet the requirements of 9.4.3.1 and 9.4.3.4 even in the presence of common mode sinusoidal voltage E_{cm} as shown in figure 9-10. The amplitude of E_{cm} 0-to-peak voltage shall be:

a) 100 V for dc to 100 kHz

b) $\frac{10}{f}$ (MHz) V for 100 kHz to 10 MHz

9.4.3.7 Transmitter fault tolerance

The transmitter shall withstand, without damage, a ±1000 V common-mode impulse applied at E_{cm}. The shape of the impulse shall be 0.3 per 50 μs (300 ns virtual front-time, 50 μs virtual time of half value) as defined in IEC Publication 60.



Note: Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-31—Transmitter impedance balance and common mode rejection

9.4.3.8 Transmitter output pulse unbalance

A single frequency power spectrum measured at the first transmitted spectrum null frequency other than dc (2.048 MHz in 4.096 Mb/s bit rate case) shall be less than –34 dB with respect to the power spectrum measured at the spectrum peak (1.024 MHz in 4.096 Mb/s bit rate case) for logical "1" input to the precoder.

Table 9-10—Frequency characteristics of 4.096 Mb/s transmit filter

Frequency (Hz)	Gain (dB)
0.2 M	0.00
0.6 M	0.04
1.0 M	–0.26
1.4 M	–0.93
2.0 M	–3.00
3.0 M	–7.83
4.0 M	–12.30
6.0 M	–19.14
8.0 M	–24.10

9.4.4 Receiver characteristics

The transmitter and the reference model circuits of medium cable (specified in 9.4.3), and a receiver to be tested shall be arranged as shown in figure 9-34 for deriving receiver characteristics. A 23 stage pseudo-random pattern shall be used as an input to the transmitter, i.e., scrambled "1."

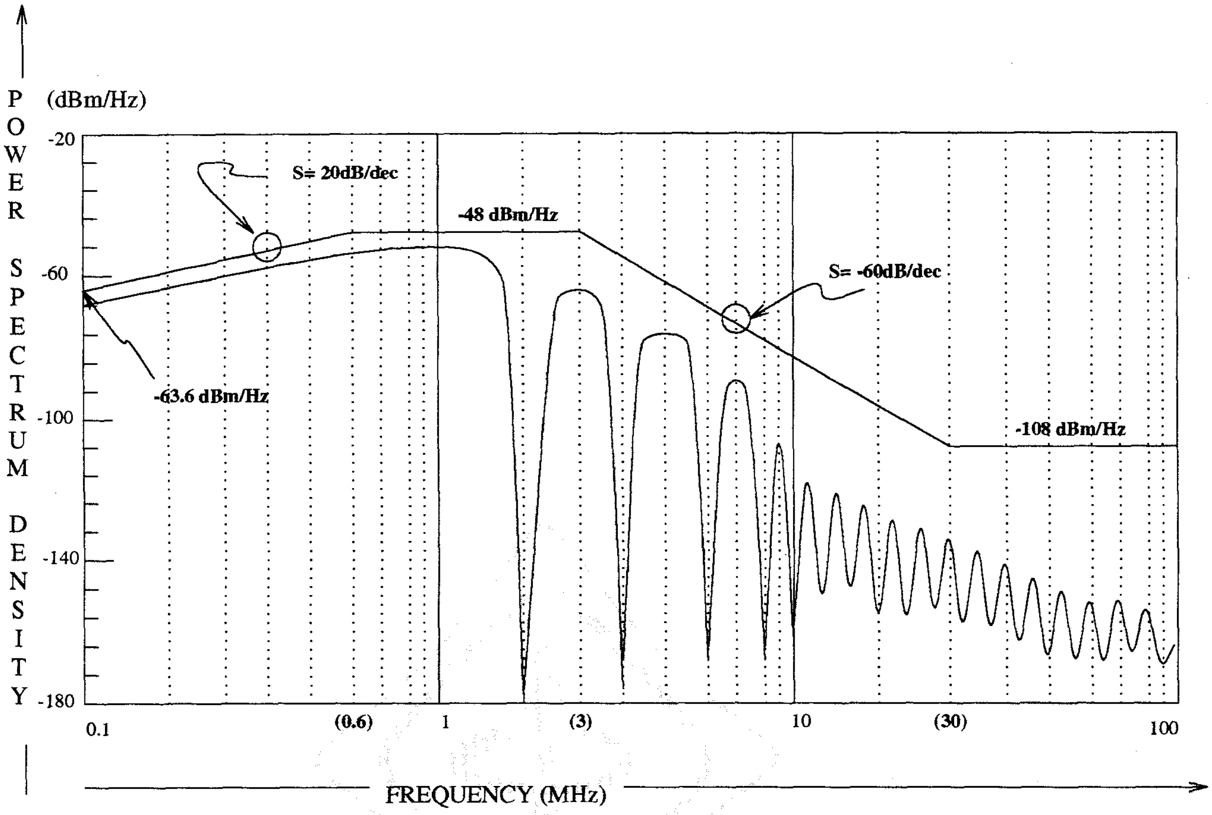


Figure 9-32—Differential output power spectrum of 4.096 Mb/s transmitter

9.4.4.1 BER requirement

The receivers shall meet the condition that the BER requirement is less than 10^{-8}. The BER is measured by the count of decoded binary 0's at the output of the decoder described in 9.4.2.3 with the frame synchronization bits excluded divided by the number of binary 1's input to the IEEE 802.9 scrambler defined in 9.3.2. The measurement period shall be long enough to count more than 10^9 bits. Detailed measuring conditions are given below.

9.4.4.2 Noise immunity

The receivers shall meet the BER requirement defined in 9.4.4 under the presence of the specified crosstalk noise, the impulsive noise, and other noises when tested separately. These noises are added to the signal at the input point of the receiver.

9.4.4.2.1 Crosstalk noise immunity

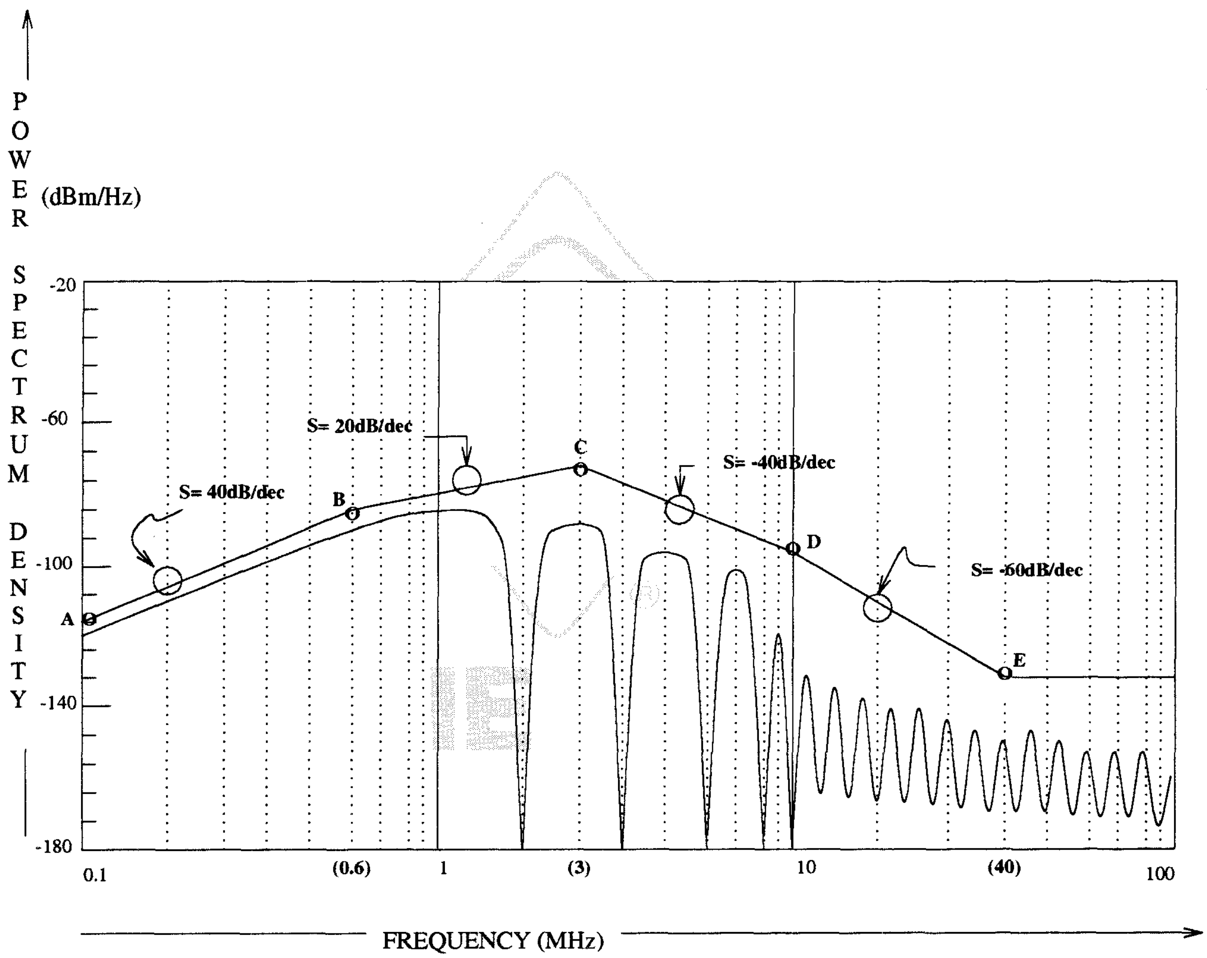
The receiver shall meet the BER requirement under the presence of simulated crosstalk noise as defined in figure 9-35. Table 9-11 lists the gain characteristics at the critical frequency points. The error rate measurement setup is defined in figure 9-36. In the absence of a transmitted signal, the applied crosstalk shall be 12 mV_{rms} at the receiver input (V_{in}) and the white noise source should have at least 10 MHz bandwidth. Note that the cable must be terminated in the effective output impedance of the transmitter in order to conduct this crosstalk measurement. The formula of frequency characteristics for the crosstalk filter is expressed by the following equation:

$$P = \left\{ k \times \frac{1}{f_o} \times \frac{\sin\left(\frac{\pi f}{2f_0}\right)}{\left(\frac{\pi f}{2f_0}\right)^2} \right\} \times \frac{1}{1 + \left(\frac{2f}{f_o}\right)^4} \times \sin^2\left(\frac{2\pi f}{f_0}\right) \times f^{\frac{3}{2}}$$

where $f_0 = 4$ MHz

9.4.4.2.2 Impulse noise immunity

The receiver shall meet the BER requirement under the presence of impulse noise as defined in 9.8.5.



AMPLITUDE AT CRITICAL POINTS:
A = -117.5 dBm/Hz
B = -86.4 dBm/Hz
C = -72.5 dBm/Hz
D = -93.4 dBm/Hz
E = -132.5 dBm/Hz

Figure 9-33—Common mode output power spectrum of 4.096 Mb/s transmitter

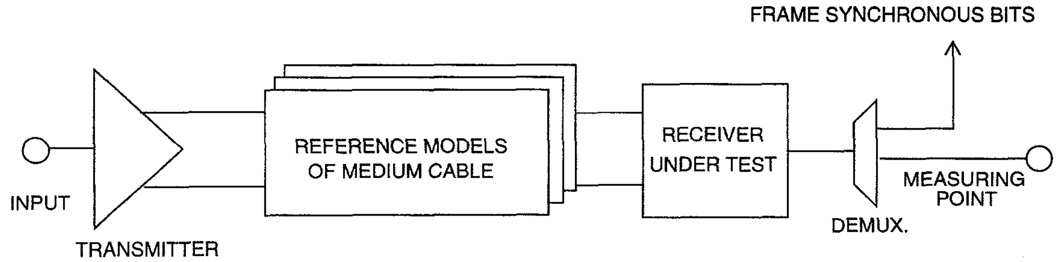


Figure 9-34—Receiver test set

9.4.4.3 Receiver input level range

The receiver shall not process an input signal whose amplitude is less than 200 mV peak-to-peak.

9.4.4.4 Receiver input impedance

The differential input impedance of the receiver, Z_R, shall be such that the reflection attenuation, defined as

$$20\log_{10}\left\{\frac{|Z_R + 100|}{|Z_R - 100|}\right\}$$

exceeds 15 dB at 1.024 MHz with a termination impedance of 100 Ω.

The common mode to differential mode impedance balance of the receiver is defined as

$$20\log_{10}\left\{\frac{E_{cm}}{E_{dif}}\right\}$$

where E_{cm} is an externally applied ac voltage as shown in figure 9-36. The impedance balance shall exceed 44 dB at 1.024 MHz.

9.4.4.5 Receiver common mode rejection

The receivers shall meet the BER requirement defined in 9.4.4 even in the presence of common mode, sinusoidal voltage E_{cm} applied as shown in figure 9-38. The amplitude of E_{cm} zero-to-peak voltage shall be:

a) 100 V for dc to 100 kHz

b) $\frac{10}{f}$ (MHz) V for 100 kHz to 10 MHz

9.4.4.6 Receiver fault tolerance

The receiver shall withstand, without damage, a ±1000 V common-mode impulse applied at E_{cm}. The shape of the impulse shall be 0.3 per 50 μs (300 ns virtual front-time, 50 μs virtual time of half value) as defined in IEC Publication 60.

9.5 PMD sublayer for higher rate (20.48 Mb/s) application

The PMD sublayer for the high rate 20.48 Mb/s applications shall use 4-point Carrierless AM/PM (4-CAP). Carrierless AM/PM is a general modulation scheme using changes in both the amplitude and phase of the signal states. The simple 4-point scheme to be used for the 20.48 Mb/s PMD sublayer uses only phase modulation. Figure 9-39 shows a block diagram of the PMD sublayer transmitter circuit and figure 9-40 shows a block diagram of the PMD sublayer receiver circuit. These diagrams are provided as models to be used in describing the various functions and one possible method of implementation. Other implementations can provide identical functionality and are line compatible.

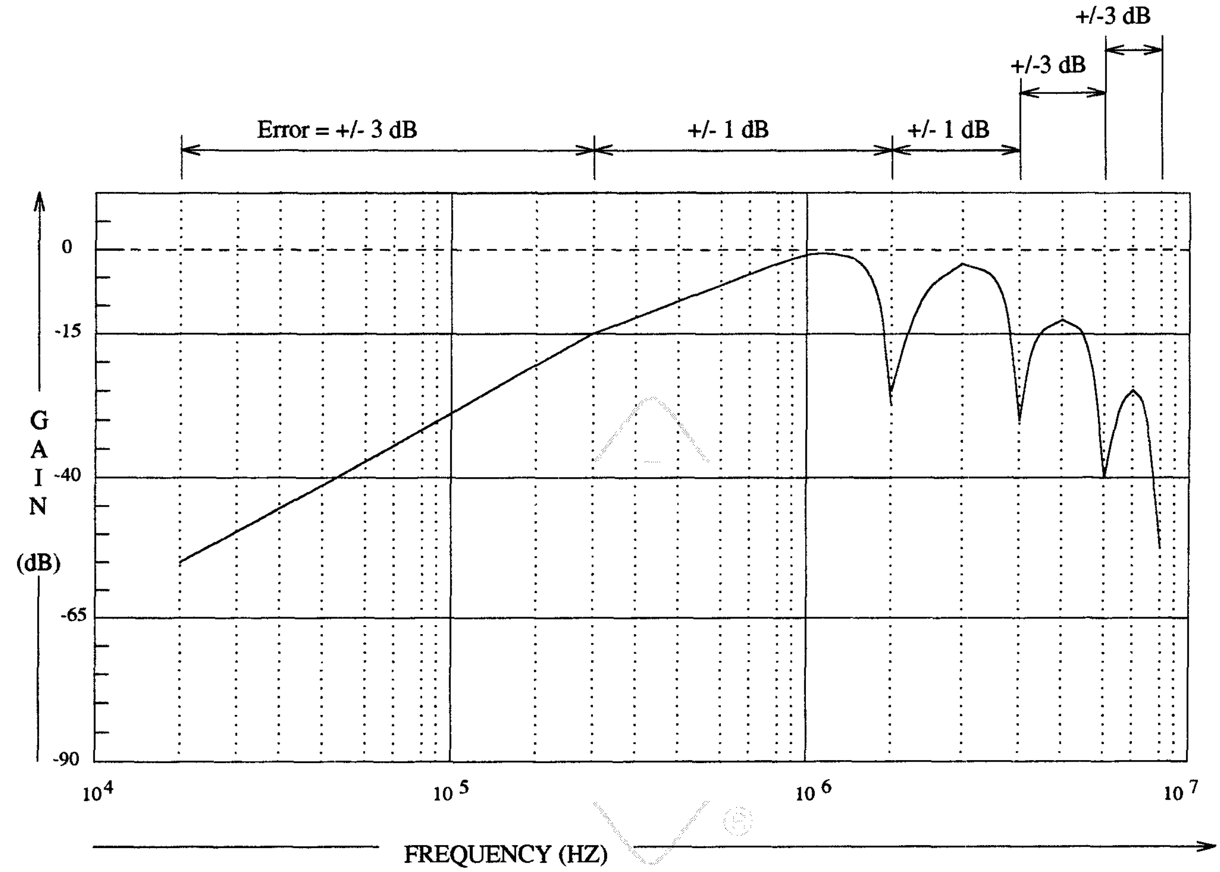


Figure 9-35—Frequency characteristics of crosstalk filter

9.5.1 Bit rate

The bit rate at the PMD sublayer/PS sublayer service boundary shall be 20.48 Mb/s. This assumes two unshielded twisted-pair wires between the ISTE (or ISTA) and the AU. The 20.48 Mb/s includes all of the required overhead fields such as those required for frame synchronization, maintenance, and parity bits. The clock rate of the AU shall be within ±32 ppm and the clock rate of the ISTE without receiving any signal from AU shall be within ±100 ppm.

9.5.2 Line code

The line code for the ISLAN interface between the PMD sublayer and the physical medium for 20.48 Mb/s shall be 4-point Carrierless AM/PM (4-CAP) operating at a symbol rate of 10.24 MBd.

Table 9-11—Frequency characteristics of crosstalk filter

Frequency (Hz)	Gain (dB)	Permitted range (dB)	
		Minimum	Maximum
20 K	–55.46	–58.46	–52.46
100 K	–31.04	–34.04	–28.04
300 K	–14.65	–15.65	–13.65
500 K	–7.52	–8.52	–6.52
1000 K	–0.41	–1.41	0.59
1160 K	–0.00	–1.00	1.00
1300 K	–0.31	–1.31	0.69
1500 K	–1.99	–2.99	–0.99
2000 K	<–20.0		
2300 K	–6.96	–7.96	–5.96
2500 K	–3.77	–4.77	–2.77
2820 K	–2.32	–3.32	–1.32
3000 K	–2.70	–3.70	–1.70
3300 K	–4.98	–5.98	–3.98
3500 K	–7.86	–8.86	–6.86
4000 K	<–20.0		
4300 K	–15.39	–16.39	–14.39
4500 K	–12.50	–13.50	–11.50
4760 K	–11.41	–12.41	–10.41
5000 K	–12.01	–13.01	–11.01
5300 K	–14.63	–15.63	–13.63
5500 K	–17.77	–18.77	–16.77
6000 K	<–20.0		
6400 K	–25.31	–26.31	–24.31
6640 K	–24.13	–25.13	–23.13
7000 K	–26.16	–27.16	–25.16
7400 K	–33.38	–34.38	–32.38
8000 K	<–20.0		

9.5.2.1 Overview

The structure of the CAP transmitter is shown in figure 9-41. The bit stream is first scrambled and then passed through an encoder, which generates real (a_n) and imaginary (b_n) symbols in each symbol period T, where the subscript n refers to symbol period nT. The encoder in figure 9-39 performs a bit-to-symbol mapping, which is explained in detail in 9.5.2.2. Each symbol (a_n and b_n) can take values equal to 0 or ±1. The four possible values taken by the complex symbols (a_n, b_n) are shown in figure 9-42.

The symbols a_n and b_n in figure 9-41 are fed to passband in-phase and quadrature shaping filters. These filters are intended to limit the bandwidth occupancy of the transmitted signal and to generate orthogonal in-phase and quadrature passband signals. The two filters have the same amplitude characteristic and phase characteristics that differ by 90°.

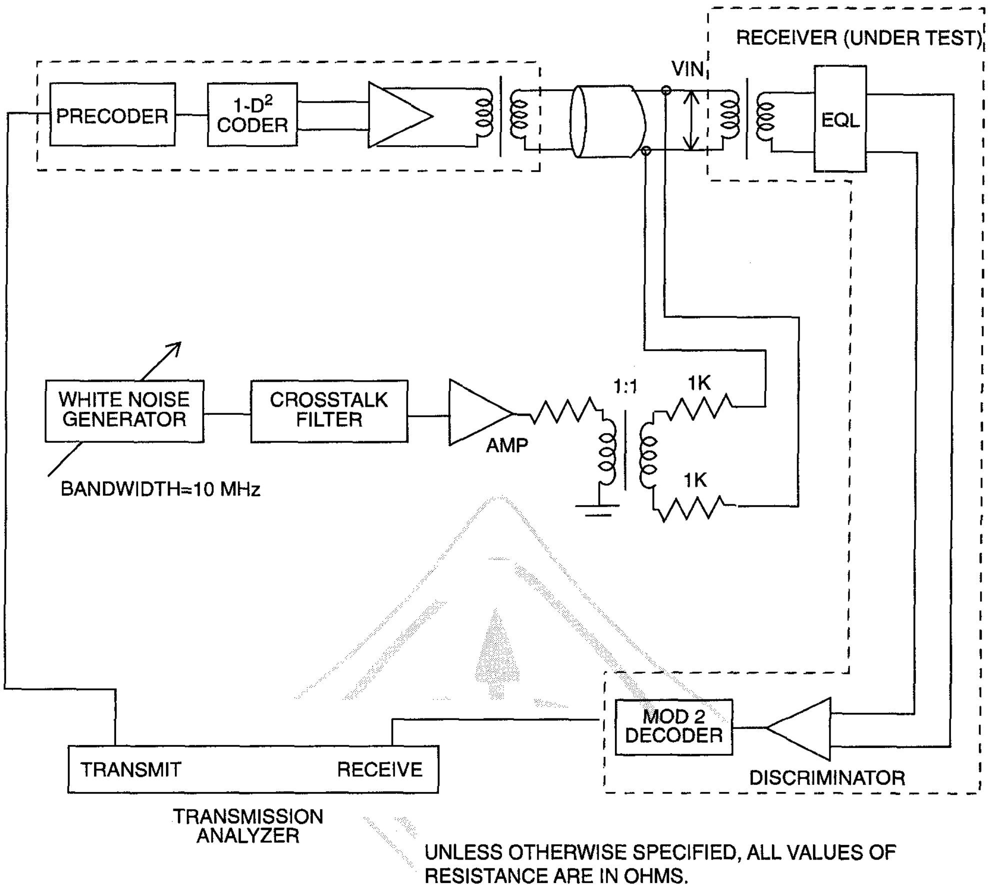


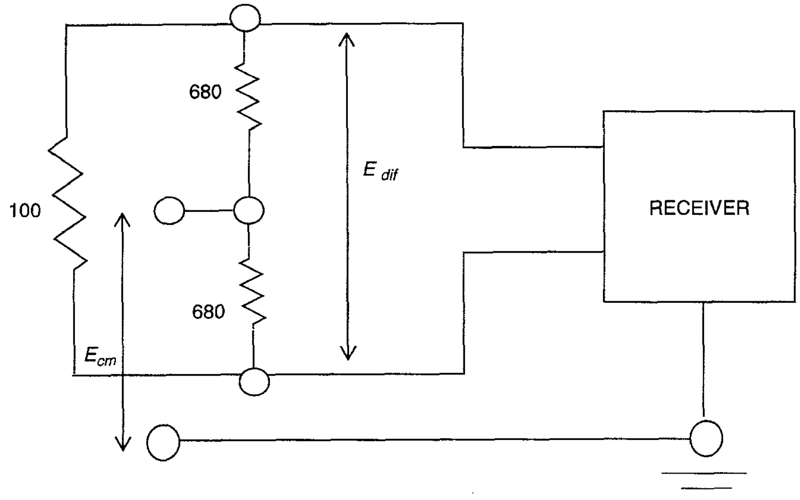
Figure 9-36—Measurement set for crosstalk noise immunity

The outputs of the two shaping filters in figure 9-41 are added and the result is fed to a low-pass filter. Where the shaping filters are realized digitally, the result would first be fed to a D/A converter, then followed by the low-pass filter.

9.5.2.2 Differential encoding and mapping

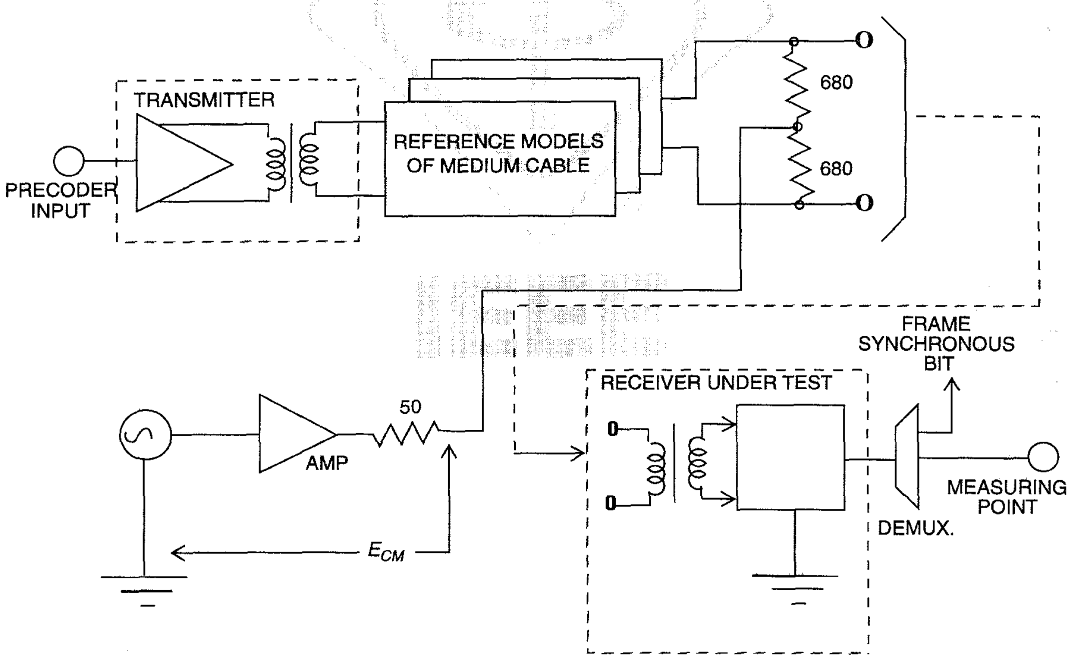
The incoming bit stream from the PS sublayer is first differentially encoded. Differential encoding is not required for reliable transmission of the 4-CAP signal since there are no phase impairments in the medium. However, to allow for possible 180° phase reversals due to medium connection errors and to allow for the possibility of more sophisticated receivers, which may introduce 90° phase shifts, a 90° differential encoding scheme shall be used.

The data stream to be transmitted shall be divided into groups of two consecutive bits, $Q1_n$ and $Q2_n$, where the subscript n designates the sequence number of the group, and the digits 1 and 2 refer to the first and second bits occurring in the data stream as they enter the encoder and mapper of the PMD sublayer, respectively. These two bits shall be differentially encoded into complex symbols (a_n, b_n) according to the signal constellation shown in figure 9-42 and as listed in table 9-12. At the receiver, the groups of two bits shall be decoded and the bits shall be reassembled in correct order.



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-37—Receiver impedance balance and common mode rejection



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-38—Common mode requirement of the receiver

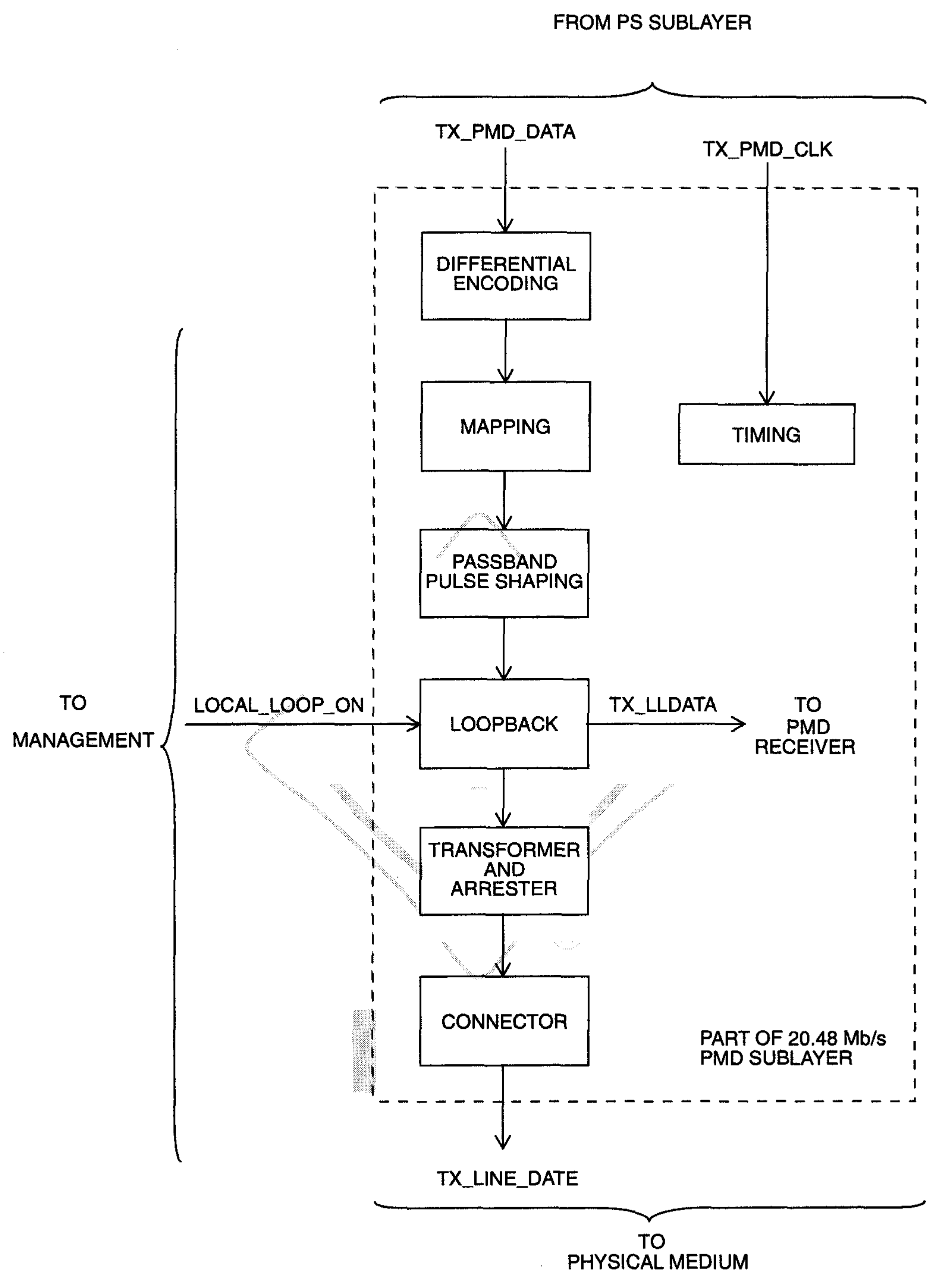


Figure 9-39—Detailed model of the transmitter for the 20.480 Mb/s PMD sublayer

9.5.2.3 Passband pulse shaping

The differentially encoded symbols shall then be passed through the passband pulse-shaping filters. These filters shall be defined as 100% excess bandwidth in-phase and quadrature passband raised cosine filters with a center frequency of 10.24 MHz and a bandwidth of 20.48 MHz.

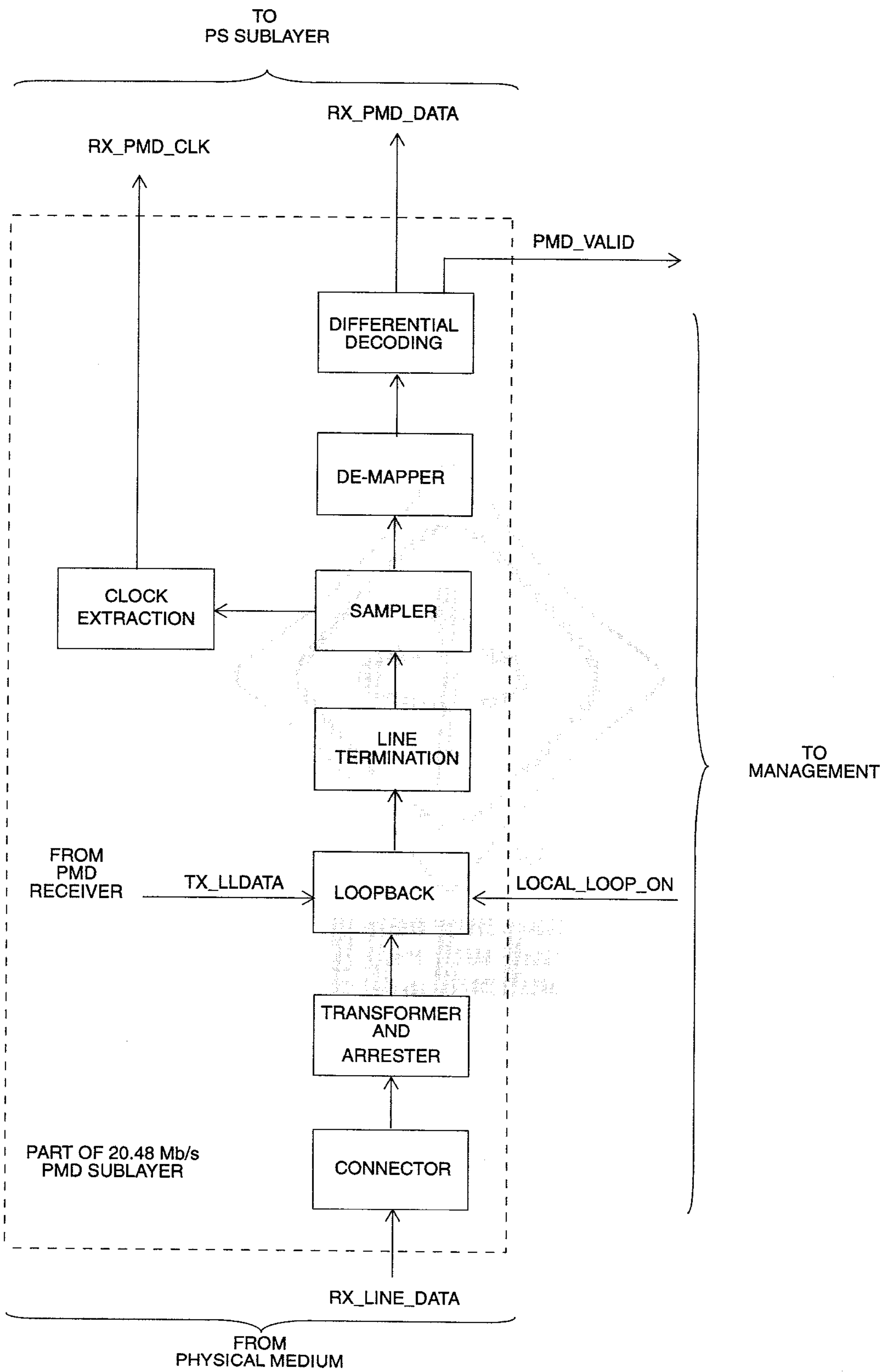


Figure 9-40—Detailed model of the receiver for the 20.480 Mb/s PMD sublayer

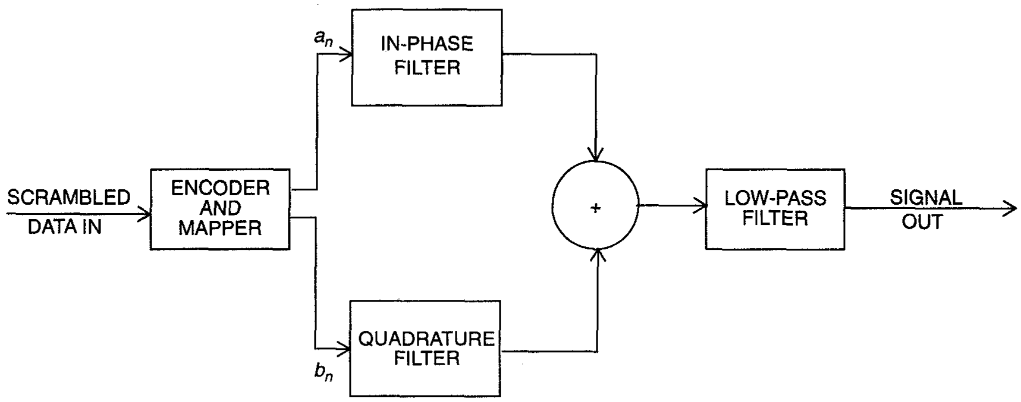


Figure 9-41—Carrierless AM/PM—Transmitter model

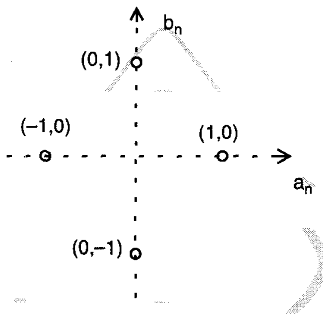


Figure 9-42—Carrierless AM/PM: receiver model

Table 9-12—4-CAP Dibit phase encoding

Inputs		Previous outputs		Phase quadrant change	Outputs	
$Q1_n$	$Q2_n$	a_{n-1}	b_{n-1}		a_n	b_n
0	0	1	0	+90°	0	1
0	0	0	1		–1	0
0	0	0	–1		1	0
0	0	–1	0		0	–1
0	1	1	0	0°	1	0
0	1	0	1		0	1
0	1	0	–1		0	–1
0	1	–1	0		–1	0
1	0	1	0	+180°	–1	0
1	0	0	1		0	–1
1	0	0	–1		0	1
1	0	–1	0		1	0
1	1	1	0	+270°	0	–1
1	1	0	1		1	0
1	1	0	–1		–1	0
1	1	–1	0		0	1

9.5.3 Transmitter characteristics

The transmitter output signal is measured at the connector of the transmitter to medium cable. All of the reference models of medium cable, which are defined in 9.5.3.1, shall be used as load to the transmitter. Unless otherwise indicated, the differential output signal and the common mode output signal shall be measured using the setups shown in figures 9-43 and 9-44, respectively.

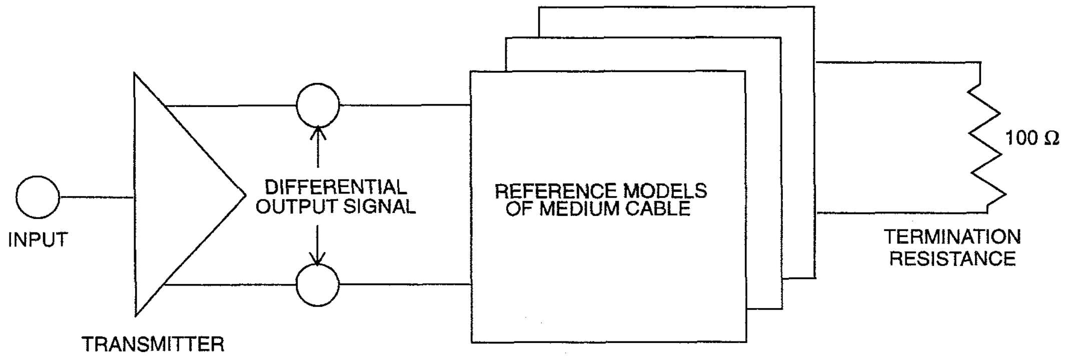
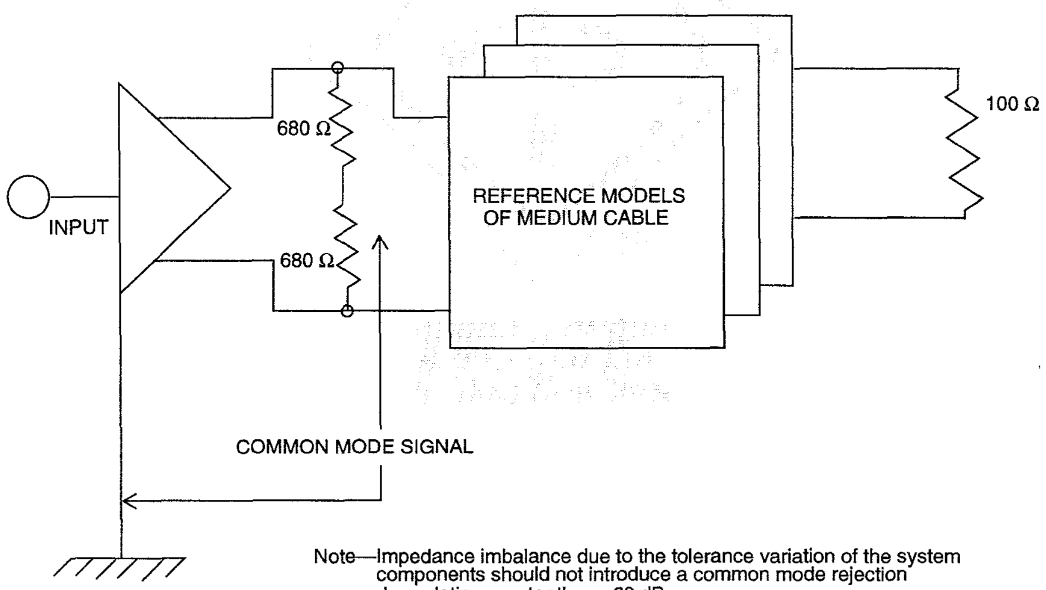


Figure 9-43—Measurement set of differential output signal



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB.

Figure 9-44—Measurement set of common mode output signal

9.5.3.1 Reference models of medium cables for 20 Mb/s data rate

The reference model circuits of medium cables are constructed as shown in figures 9-45 and 9-46. The higher rate PMD is affected by the horizontal subsystem of reach up to 135 m. Consequently, the two

Juniper Ex 1032 - 208
Juniper v Chrimar
IPR2016-01399

reference lengths used are 50 m and 100 m. The circuit illustrated in figure 9-45 is a model of a 50 m length of 24-gauge D-type inside wire (DIW) and figure 9-46 is a model of a 100 m length of 24-gauge DIW.

Unless otherwise indicated, the value of resistance is in ohms (Ω), the value of inductance is in microhenries (μH), and the value of capacitance is in farads (F). The direction of signal flow is from left to right.

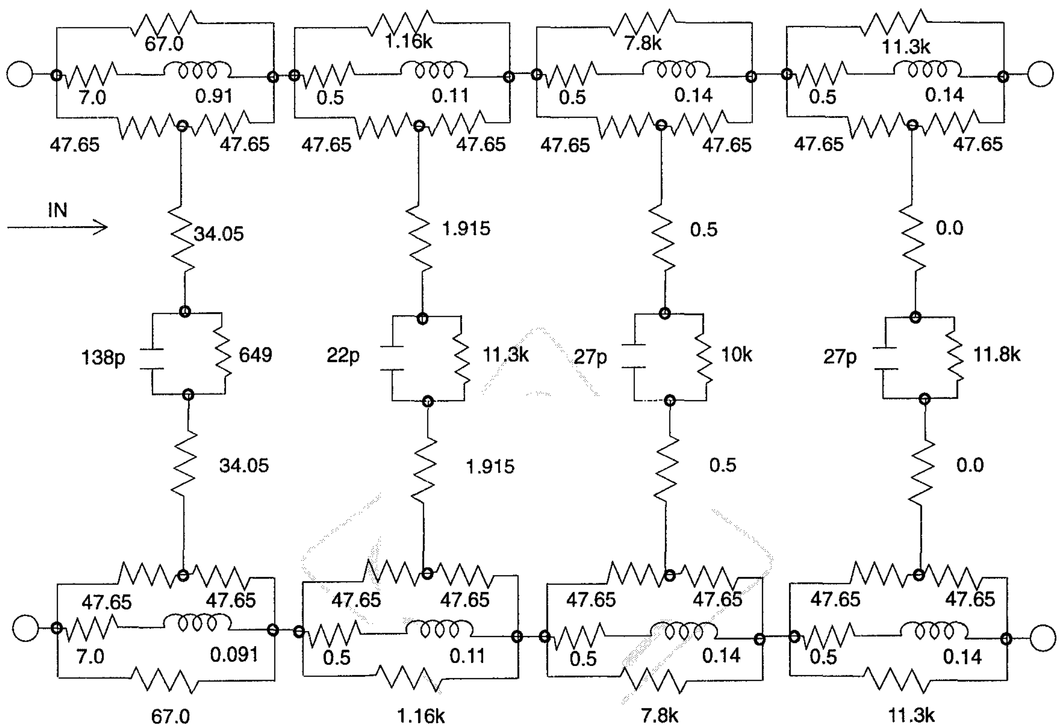


Figure 9-45—Reference model of 50 m medium cable

9.5.3.2 Transmitter output signal spectrum

The frequency characteristics of the transmit filter are shown in figure 9-47. Table 9-13 lists the gain characteristics at the critical frequency points. The transmit filter is a seventh-order, elliptic, low-pass filter whose frequency cutoff is at 20 MHz. The errors of the cutoff frequency and the Q-value should be less than 10%. The power spectrum of the differential output signal and that of the common mode output signal for a pseudo-random sequence shall be within the mask shown in figure 9-48. A pseudo-random sequence can be generated by applying binary 1's to the IEEE 802.9 scrambler specified in 9.3.2.

9.5.3.3 Transmitter amplitude

The amplitude of the differential output signal is 3 V peak-to-peak.

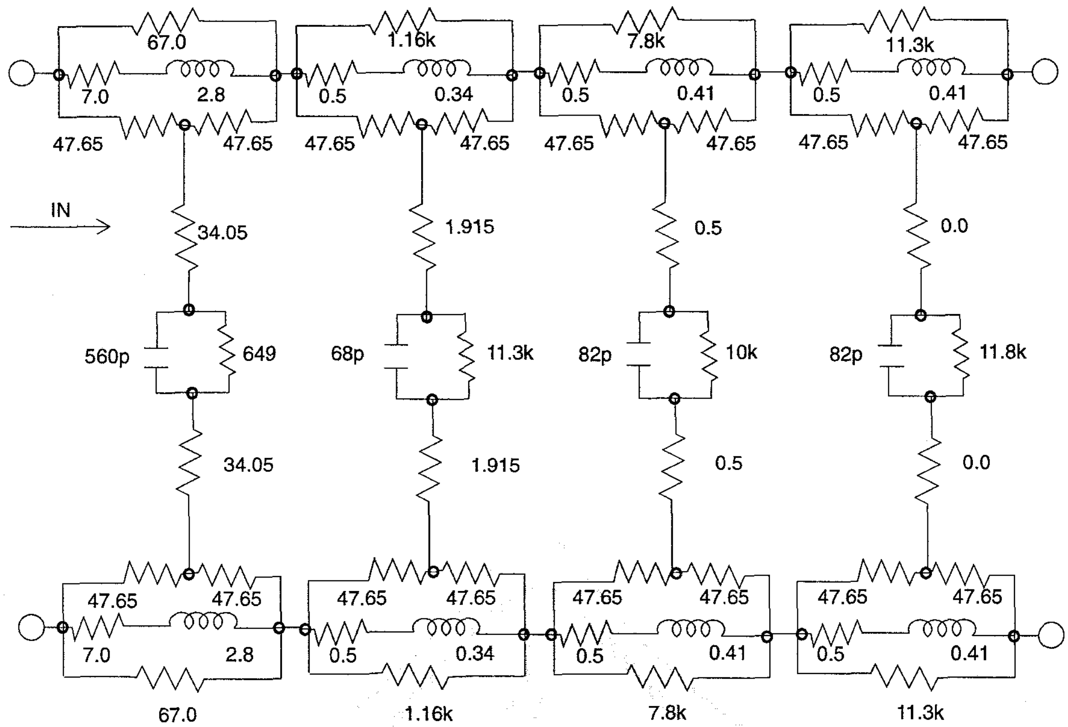


Figure 9-46—Reference model of 100 m medium cable

9.5.3.4 Transmitter output impedance

The differential output impedance of the transmitter, Z_T, shall be such that the reflection attenuation, defined as

$$20\log_{10}\left\{\frac{|Z_T + 100|}{|Z_T - 100|}\right\}$$

exceeds 15 dB at 10.24 MHz with 100 Ω termination.

The common mode to differential mode impedance balance is defined as

$$20\log_{10}\left\{\frac{E_{cm}}{E_{diff}}\right\}$$

where E_{cm} is an externally applied ac voltage as shown in figure 9-49. The impedance balance shall exceed 44 dB at 10.24 MHz.

9.5.3.5 Transmitter output timing jitter

The transmitter output jitter shall be no more than ±10 ns.

9.5.3.6 Transmitter common mode requirement

The transmitter shall meet the requirements of 9.5.3.2 even in the presence of common mode sinusoidal voltage, E_{cm}, as shown in figure 9-49. The amplitude of E_{cm} zero-to-peak voltage shall be:

a) 100 V for dc to 100 kHz

b) $\frac{10}{f}$ (MHz) V for 100 kHz to 10 MHz

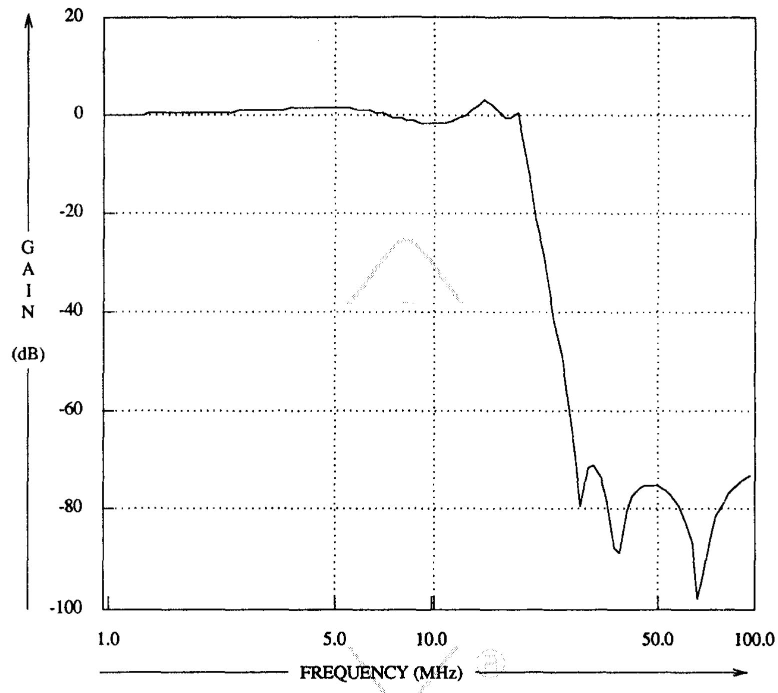


Figure 9-47—Frequency characteristics of transmit filter for 20.48 Mb/s

9.5.3.7 Transmitter fault tolerance

The transmitter shall withstand, without damage, a ±1000 V common-mode impulse applied at E_{cm}. The shape of the impulse shall be 0.3 per 50 μs (300 ns virtual front-time, 50 μs virtual time of half value) as defined in IEC Publication 60.

9.5.4 Receiver characteristics

The transmitter and the reference model circuits of medium cable (both defined in 9.5.3), and a receiver to be tested are arranged as shown in figure 9-50 for deriving receiver characteristics. A 23-stage pseudo-random pattern shall be used as input to the transmitter, i.e., scrambled "1" using the scrambler of 9.3.3.

9.5.4.1 BER requirement

The receiver shall have a BER requirement of less than 10^{-8}. The BER is measured by the count of decoded and descrambled binary 0's at the output of the descrambler described in 9.3.2, with the frame synchronization bits excluded divided by the number of binary 1's input to the IEEE 802.9 scrambler defined in 9.3.3. The measurement period shall be long enough to count more than 10^9 bits. Detailed measuring conditions are given in 9.5.4.2 through 9.5.4.6.

Table 9-13—Frequency characteristics of 20.48 Mb/s transmit filter

Frequency (MHz)	Gain (dB)
1.00	0.13
1.15	0.18
1.32	0.23
1.51	0.30
1.74	0.39
2.00	0.50
2.29	0.65
2.63	0.82
3.02	1.03
3.47	1.25
3.98	1.45
4.57	1.55
5.25	1.46
6.03	1.04
6.92	0.29
7.94	–0.65
9.12	–1.48
10.47	–1.74
12.02	–0.73
13.80	2.26
15.85	0.84
18.20	0.47
20.89	–21.95
23.99	–41.74
27.54	–68.78
31.62	–71.30
36.31	–87.99
41.69	–77.35
47.86	–75.07
54.95	–77.70
63.10	–87.04
72.44	–85.67
83.18	–76.83
95.50	–72.96

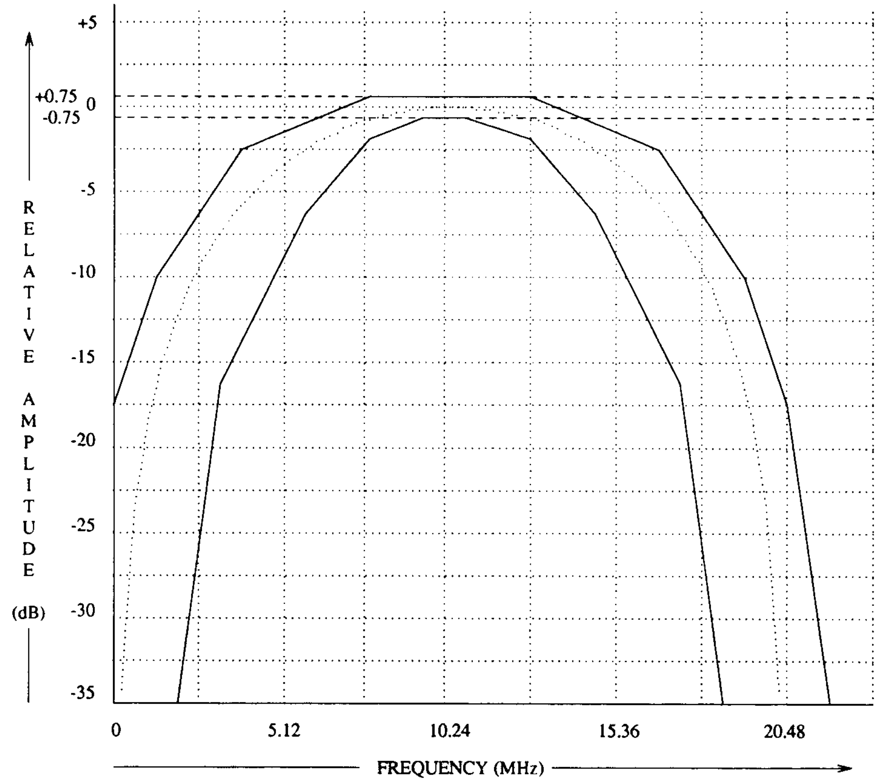


Figure 9-48—Differential output power spectrum of 20.48 Mb/s transmitter

9.5.4.2 Noise immunity

The receiver shall meet the BER requirement defined in 9.5.4.1 under the presence of the specified crosstalk noise, the impulsive noise, and other noises. These noises are added to the signal at the input point of the receiver.

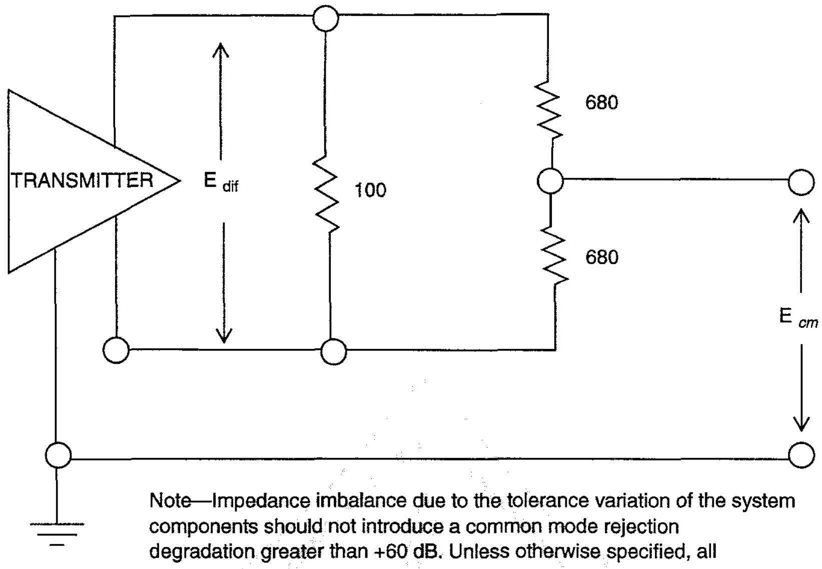
9.5.4.2.1 Crosstalk noise immunity

The receiver shall meet the BER requirement under the presence of simulated crosstalk noise defined in figure 9-23. Table 9-14 lists the gain characteristics at the critical frequency points. The error rate measurement setup is defined in figure 9-52. In the absence of a transmitted signal, the applied crosstalk shall be 150 mV_{rms} at the receiver input (V_{in}) and the white noise source should have at least 20 MHz bandwidth. Note that the cable must be terminated in the effective output impedance of the transmitter in order to conduct this crosstalk measurement.

The formula for the squared magnitude of the amplitude characteristic of the crosstalk filter is expressed by the following equation:

$$X^2(f) = \frac{k}{4f_c^2} \times \left[1 + \sin\frac{\pi}{f_c}\left(f - \frac{f_c}{2}\right)\right]^2 \times |G(f)|^2 \times f^{\frac{3}{2}} \qquad 0 \le f \le 30MHz$$

where f_c = 10 MHz and G (f) is the transfer function of the low-pass filter described in table 6-24 and table 9-13. The amplitude characteristic of the crosstalk filter's transfer function is given in figure 9-51 and table 9-14. This amplitude characteristic follows the above expression up to 30 MHz. For higher frequencies, the gain of the filter shall be at least 70 dB below the maximum gain.



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-49—Transmitter impedance balance and common mode rejection

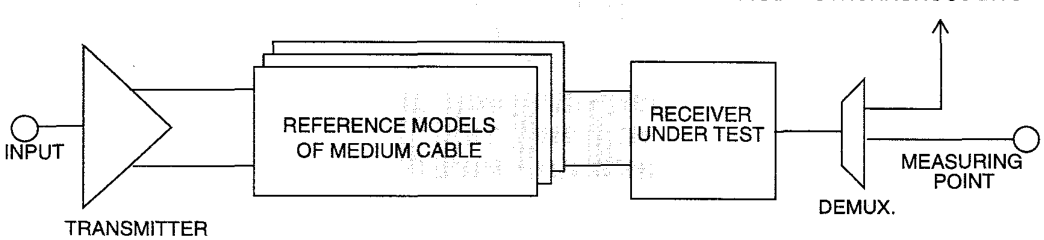


Figure 9-50—Receiver test set

9.5.4.2.2 Impulse noise immunity

The receiver shall meet the BER requirement, defined in 9.5.4.1, under the presence of impulse noise as defined in 9.8.5 and 9.9.5.

9.5.4.3 Receiver input level range

The receiver shall not process an input signal whose amplitude is less than 300 mV peak-to-peak.

9.5.4.4 Receiver input impedance

The differential input impedance of the receiver, Z_R, shall be such that the reflection attenuation, defined as

$$20\log_{10}\left\{\frac{|Z_R+100|}{|Z_R-100|}\right\}$$

exceeds 15 dB at 10.24 MHz with a termination impedance of 100 Ω.

Table 9-14—Frequency characteristics of 20.48 Mb/s crosstalk filter

Frequency (MHz)	Gain (dB)	Permitted range	
		Max	Min
0	–84.33	–81.33	–87.33
1	–54.84	–51.84	–57.84
2	–36.03	–33.03	–39.03
3	–25.29	–22.29	–28.29
4	–17.97	–14.97	–20.97
5	–12.60	–11.60	–13.60
6	–8.54	–7.54	–9.54
7	–5.45	–4.45	–6.45
8	–3.14	–.14	–4.14
9	–1.51	–0.51	–2.51
10	–0.47	0.53	–1.47
11	0.00	1.00	–1.00
12	–0.10	0.90	–1.10
13	–0.81	0.19	–1.81
14	–2.18	–1.18	–3.18
15	–4.36	–3.36	–5.36
16	–7.57	–6.57	–8.57
17	–12.28	–11.28	–13.28
18	–19.55	–16.55	–22.55
19	–32.76	–29.76	–35.76
20	–84.33	–81.33	–87.33
21	–41.22	–38.22	–44.22
22	–36.46	–33.46	–39.46
23	–37.64	–34.46	–40.64
24	–41.37	–38.37	–44.37
25	–46.58	–43.58	–49.58
26	–52.80	–49.80	–55.80
27	–59.80	–56.80	–62.80
28	–67.43	–64.43	–70.43
29	–75.62	–72.62	–78.62
30	–84.33	–81.33	–87.33

The common mode to differential mode impedance balance of the receiver is defined as

$$20\log_{10}\left\{\frac{E_{cm}}{E_{diff}}\right\}$$

where E_{cm} is an externally applied ac voltage as shown in figure 9-53.

The impedance balance shall exceed 44 dB at 10.24 MHz.

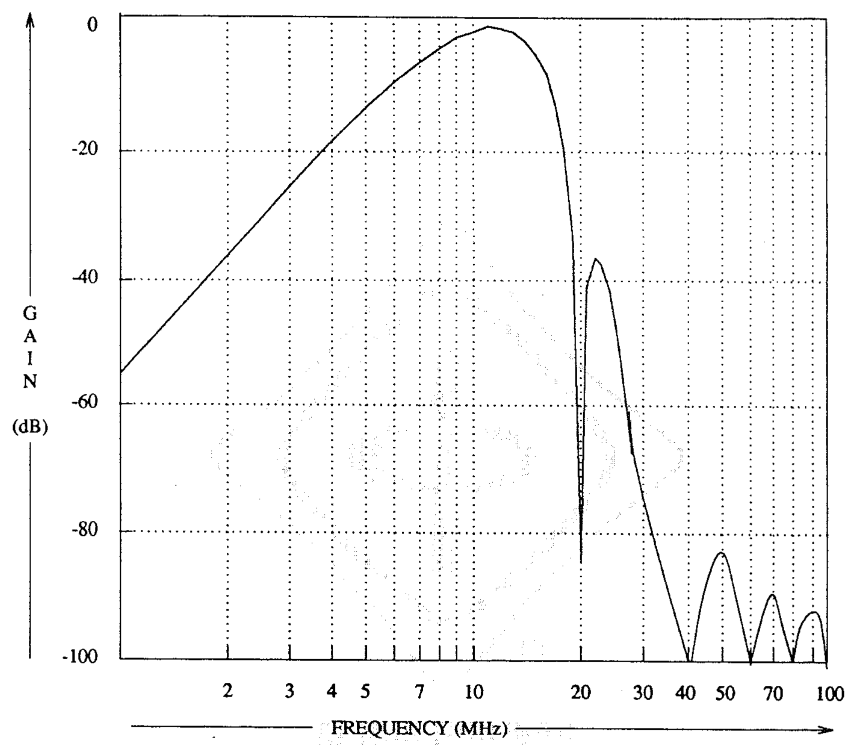


Figure 9-51—Frequency characteristics of 20.48 Mb/s crosstalk filter

9.5.4.5 Receiver common mode rejection

The receiver shall meet the BER requirement defined in 9.5.4 even in the presence of common mode sinusoidal voltage E_{cm} applied as shown in figure 9-54. The amplitude of E_{cm} zero-to-peak voltage shall be:

a) 100 V for dc to 100 kHz

b) $\frac{10}{f}$ (MHz) V for 100 kHz to 10 MHz

9.5.4.6 Fault tolerance

The receiver shall withstand, without damage, a ±1000 V common-mode impulse applied at E_{cm}. The shape of the impulse shall be 0.3 per 50 μs (300 ns virtual front-time, 50 μs virtual time of half value) as defined in IEC Publication 60.

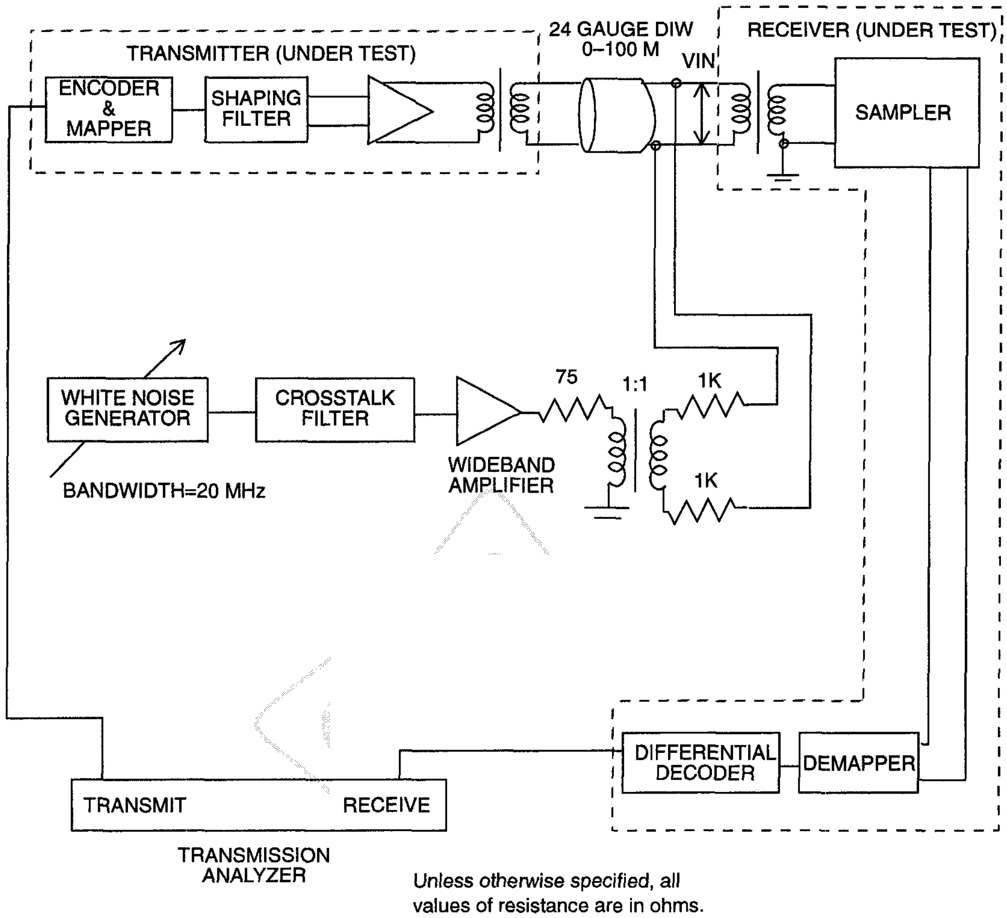


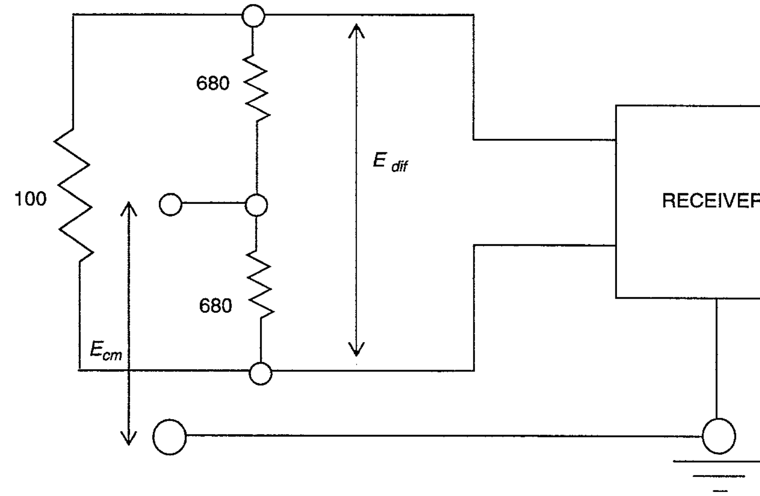
Figure 9-52—Measurement set for crosstalk noise immunity

9.6 Connector

The recommended connector is the 8-position (pole) modular jack. This connector, shown in figure 9-55, has been previously specified by the ITU-T for the ISDN basic rate interface and by ISO/IEC 8802-3 for the 10BASE-T interface. The mechanical characteristics of this connector are specified in ISO/IEC 8877:1992. It is recognized that another implementation of the AU could be a multiport device and other connector types may be appropriate for this environment. However, even under these circumstances, the ISTE connector shall be the 8-position modular jack. In the industry nomenclature, this jack is referred to as an RJ45-type.

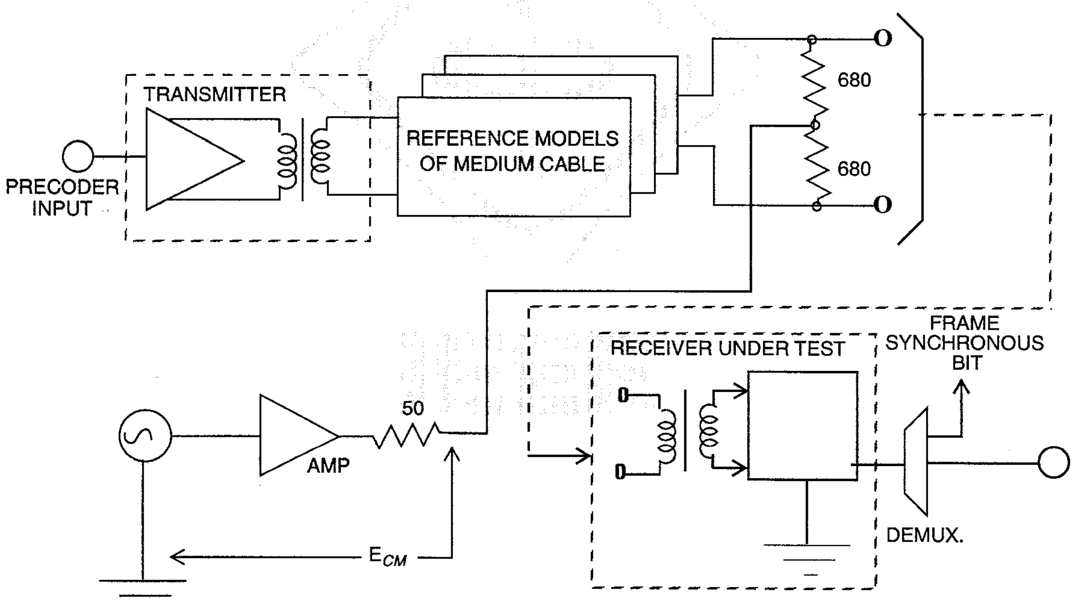
9.6.1 Pin-pair assignments

The recommended grouping of pins as functional pairs for the ISTE and AU is shown in figure 9-55. Also note that provision has been made for powering across the interface. See 9.7 for further discussion of this issue.



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +60 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-53—Receiver impedance balance and common mode rejection



Note—Impedance imbalance due to the tolerance variation of the system components should not introduce a common mode rejection degradation greater than +70 dB. Unless otherwise specified, all values of resistance are in ohms.

Figure 9-54—Common mode requirement of the receiver

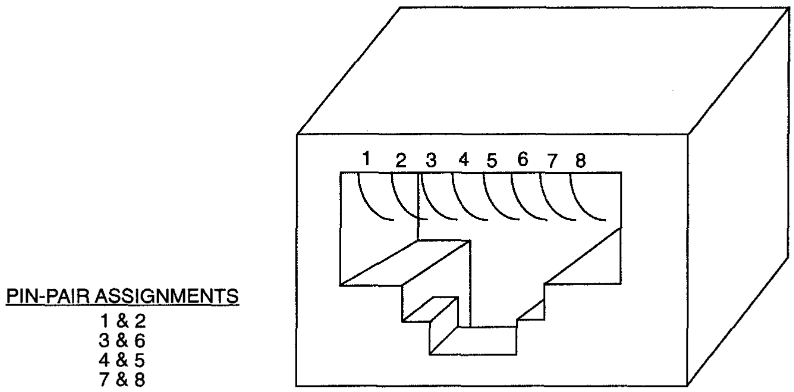


Figure 9-55—Recommended connector

4, 5	NOT USED
1, 2	ISTE TRANSMIT PAIR
3, 6	ISTE RECEIVE PAIR
7, 8	RESERVED

Figure 9-56—Recommended pin-pair assignments

9.6.2 Functional assignments

As shown in figure 9-56, pins 1 & 2 of the modular jack have been designated as ISTE transmit and pins 3 & 6 have been designated as ISTE receive. In addition, pins 7 & 8 have been reserved for powering applications.

9.7 Powering considerations

Interface powering of the ISTE is not a requirement of this standard, but provision of connector pins has been made for this purpose. Specifically, pins 7 & 8 have been reserved for the purpose of remote power feeding of the ISTE. In addition, the use of phantom powering (using pin-pairs 1 & 2 and 3 & 6) is not precluded, but is not included in this standard.

Any further work on powering should, to the extent possible, maintain consistency with that done for the ISDN basic rate interface in CCITT Recommendation I.430 (1993).

9.8 Medium cable characteristics—4.096 Mb/s PMD

9.8.1 Definition and maximum length of the medium cable

The medium cable for the IEEE 802.9 ISLAN interface for the 4.096 Mb/s line rate should be unshielded telephone twisted pairs (UTTP). Both horizontal and backbone wiring are defined in ANSI/EIA/TIA 568-91 as 100 Ω horizontal and backbone UTTP. The maximum length of a set of horizontal and backbone wiring shall be 450 m.

The length of applicable cable may be restricted by the result of the testing defined in 9.8.2.

9.8.2 Medium transmission characteristics—4.096 Mb/s PMD

Medium cables shall meet the condition that the eye opening through at least one of the standard equalizers is greater than 70%. In the measurement, the transmitter (defined in 9.4.3), the medium cable, and a standard equalizer are arranged as shown in figure 9-57.

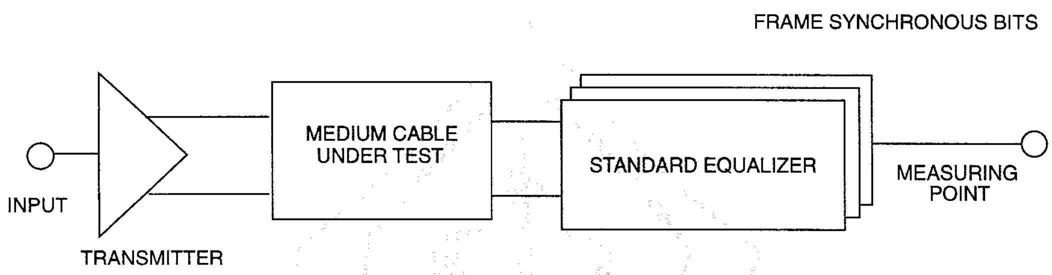


Figure 9-57—Medium cable test set-4.096 Mb/s PMD

The characteristics of three different types of the standard equalizer are defined by the following equation:

$$A = \left| \frac{\left(1 + \frac{jf}{f_1}\right)}{\left(1 + \frac{jf}{f_2}\right) \times \left(1 + \frac{jf}{f_3}\right)^2} \right|$$

The three types of equalizers have the following frequency coefficients:

Type	f_1	f_2	f_3
150 m	700 kHz	1.7 MHz	6.9 MHz
300 m	550 kHz	3.5 MHz	9.3 MHz
450 m	350 kHz	8.5 MHz	9.9 MHz

The error of cutoff frequency should be less than 5%. Figures 9-58 through 9-60, and table 9-15 show the frequency characteristics of three different types of standard equalizer for cable medium tests to be conducted against the 4.096 Mb/s PMD. Figure 9-58 corresponds to the frequency characteristics for a 150 m length of DIW. Figure 9-59 corresponds to the frequency characteristics for a 300 m length of DIW. Figure 9-60 corresponds to the frequency characteristics for a 450 m length of DIW.

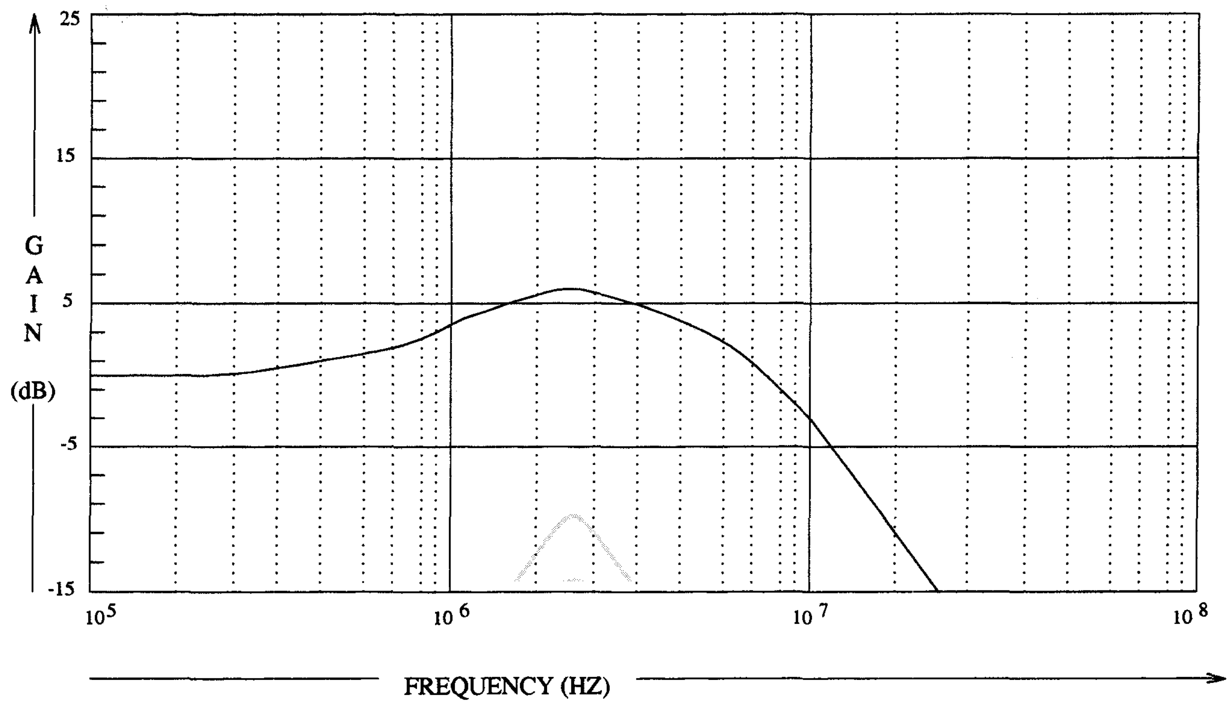


Figure 9-58—Frequency characteristics of standard equalizer at 150 m for 4.096 Mb/s

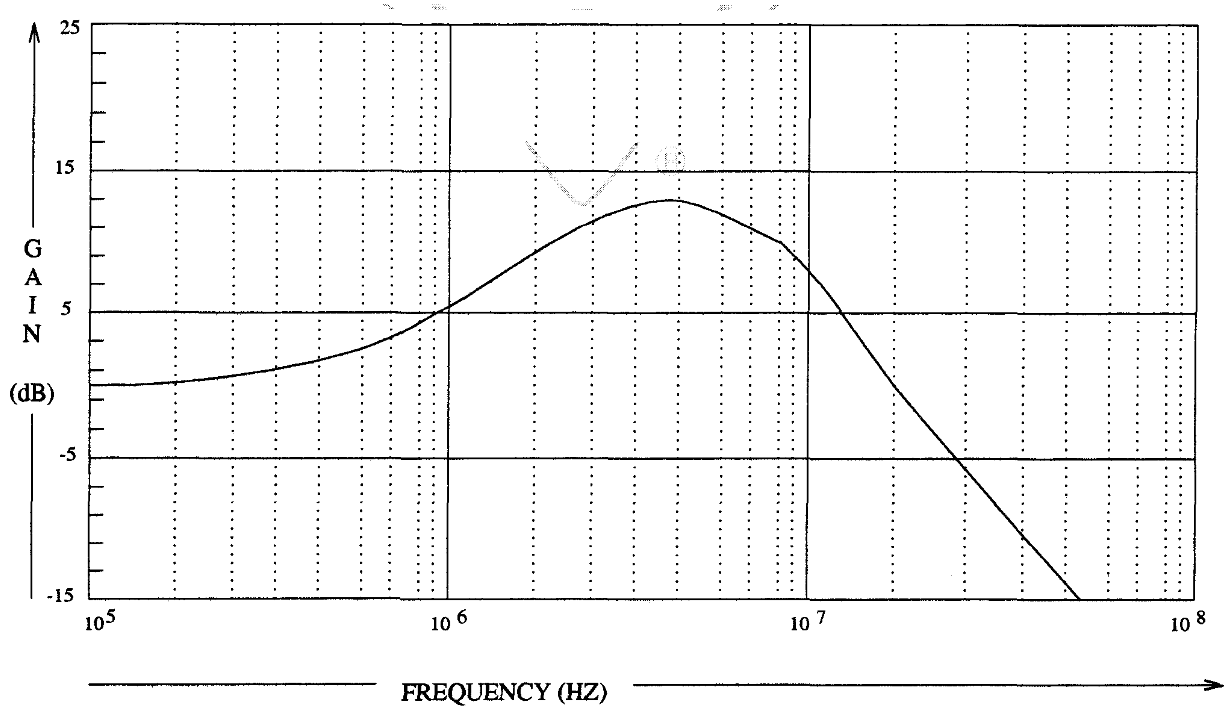


Figure 9-59—Frequency characteristics of standard equalizer at 300 m for 4.096 Mb/s

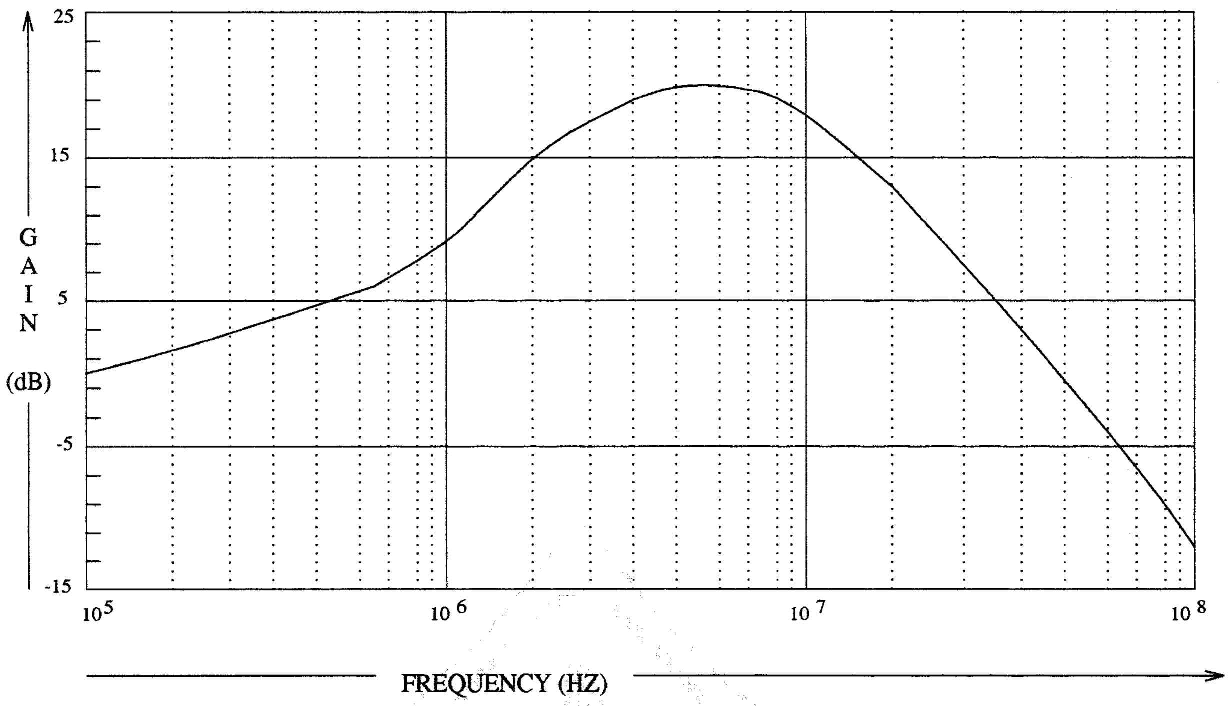


Figure 9-60—Frequency characteristics of standard equalizer at 450 m for 4.096 Mb/s

9.8.3 Medium characteristics impedance—4.096 Mb/s PMD

The commercial cable model used as a reference for this standard recommends that the magnitude of the differential characteristics impedance at 1.024 MHz of the medium cable shall be 100 Ω ±15% as defined in ANSI/EIA/TIA 568-91.

9.8.4 Crosstalk noise—4.096 Mb/s PMD

It is recommended that the pair-to-pair and multiple disturber crosstalk attenuation shall be at the minimum values defined in ANSI/EIA/TIA 568-91.

9.8.5 Impulse noise—4.096 Mb/s PMD

The noise voltage on wire pairs terminated at both ends in 100 Ω, as measured through the following specified filters, shall not exceed the corresponding threshold voltages more than 9 times per 1800 s interval. Following the start of any particular impulse that is counted, any additional impulse shall be ignored (i.e., not be counted) for a period of 100 μs. Each filter is a 2-pole Butterworth low-pass filter with the indicated cutoff (3 dB point) frequency, as defined by Section 12 of ISO/IEC 8802-3:1993.

Cutoff frequency	Threshold
2 MHz	170 mV
4 MHz	275 mV
10 MHz	560 mV

Table 9-15—Frequency characteristics of standard equalizer for 4.096 Mb/s

Frequency	Gain (dB)		
	150 m	300 m	450 m
200 K	0.274	0.52	1.22
600 K	1.82	3.24	5.90
1000 K	3.36	5.90	9.47
2000 K	5.15	9.91	14.70
3000 K	5.22	11.60	17.40
4000 K	4.60	12.20	19.00
5000 K	3.65	12.20	19.90
6000 K	2.54	11.80	20.20
7000 K	1.35	11.20	20.30
8000 K	1.48	10.50	20.10
9000 K	–1.05	9.74	19.70
10000 K	–2.22	8.91	19.20
11000 K	–3.36	8.07	18.70
12000 K	–4.46	7.21	18.10

The impulse noise occurrence rate changes inversely by one decade for each 7 dB change in the threshold voltage. If a count rate of N counts per 1800 s period is measured on a specific cable and filter at the specified voltage threshold, the media noise margin is $7 \log_{10} (9/N)$ dB.

9.9 Medium cable characteristics—20.48 Mb/s PMD

9.9.1 Definition and maximum length of the medium cable

The medium cable for the IEEE 802.9 ISLAN interface for the 20.480 Mb/s line rate should be UTTP. Only horizontal wiring is defined for this application in ANSI/EIA/TIA 568-91 as 100 Ω horizontal UTTP. The maximum length shall be 135 m.

The length of applicable cable may be restricted by the result of the testing defined in 9.9.2.

9.9.2 Medium transmission characteristics—20.48 Mb/s PMD

Medium cables shall meet the condition that the eye opening through the transceiver is greater than 70%. In the measurement, the transmitter (defined in 9.5.3), and the medium cable are arranged as shown in figure 9-61. No equalization is required for the 20.48 Mb/s PMD.

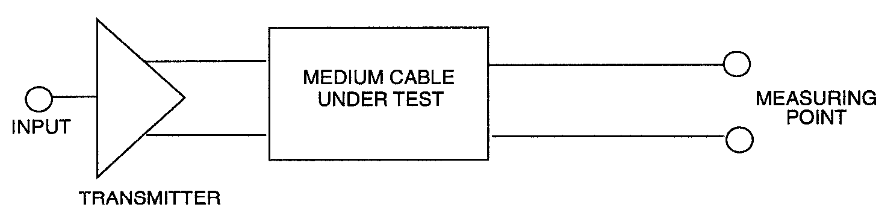


Figure 9-61—Medium cable test set—20.48 Mb/s

9.9.3 Medium characteristics impedance—20.48 Mb/s PMD

The magnitude of the differential characteristics impedance at 1.024 MHz of the medium cable shall be 100 ±15% as defined in ANSI/EIA/TIA 568-91.

9.9.4 Crosstalk noise—20.48 Mb/s PMD

It is recommended that the pair-to-pair and multiple disturber crosstalk attenuation shall be at the minimum values defined in ANSI/EIA/TIA 568-91.

9.9.5 Impulse noise—20.48 Mb/s PMD

The noise voltage on wire pairs terminated at both ends in 100 Ω, as measured through the following specified filters, shall not exceed the corresponding threshold voltages more than 0.2 times per second. Following the start of any particular impulse that is counted, any additional impulse shall be ignored (i.e., not be counted) for a period of 1 μs. Each filter is a 3-pole Butterworth low-pass filter with the indicated cutoff (3 dB point) frequency, as defined in ISO/IEC 8802-3:1993.

The implementation of the 3-pole Butterworth low-pass filter shall have the characteristics in table 9-16.

Table 9-16—Filter characteristics

Characteristic	Value
Cutoff frequency (3 dB)	15 MHz
Insertion loss (5–10 MHz)	≤ 1.0 dB
Attenuation (30 MHz)	17.5 dB min.
Input impedance (5-10 MHz)	100 Ω
Return loss with 100 Ω load (5–10 MHz)	≥ 20 dB

The impulse noise occurrence rate changes inversely by one decade for each 7 dB change in the threshold voltage. If a count rate of N counts per 1 s period is measured on a specific cable and filter at the specified voltage threshold, the media noise margin is $7 \log_{10} (0.2/N)$ dB.

10. ISLAN layer management

10.1 Introduction

The specification of the management functions at the integrated services (IS) local area network (LAN) interface involves the definition of ISLAN managed objects (MOs), with their attributes, operations, and notifications, as required by management applications specific to the integrated transmission services and application environment. However, in all management specifications, the management of the ISLAN shall be fully conformant with operating systems interconnection (OSI) standards for systems and layer management. Since the ISLAN interface also provides access to IS digital network (ISDN) bearer services and teleservices, the management of the interface shall also be conformant to standards defined for the management of the ISDN user/network interface.

10.2 Characteristics of the ISLAN interface relevant to management

It is important to understand the distinctive characteristics of the ISLAN environment that are relevant to its management. The following is a list of the most significant points.

a) The ISLAN is, at the same time, an IEEE 802 LAN, a local access arrangement for connection to a variety of backbone network technologies [e.g., an IEEE 802.6 metropolitan area network (MAN)], and an ISDN local access interface. Therefore, ISLAN management shall coordinate the management requirements of the various architectures of its local access services.

   The management of the communication between an ISTE and any other open system, whether the latter is located on the same ISLAN, or on some backbone network, or on some remote subnetwork, shall encompass both the scenario in which the access unit (AU) is simply used as a local access bridge to other networks and the scenario in which the AU is used to provide local LAN services for its connected IS terminal equipment (ISTE).

b) The AU shall play the role of a packet-switching and circuit-switching server for the information flow to or from the ISTEs attached to it. Consequently, the AU has management functions that monitor and control the functioning of the ISTEs.

c) The ISTE is an information flow and control termination for IEEE 802 LAN services, ISDN bearer services, and possibly a variety of other backbone LAN services. This means that the ISTE shall meet the standard management requirements of each of the types of terminal architectures it embodies.

d) The ISTE has a hybrid architecture. Above the PHY service boundary, the ISTE shall support at least three types of protocol stacks:

   1) An OSI standard protocol stack above the IEEE 802.9 medium access control (MAC) service boundary, beginning with the logical link control (LLC) entity.

   2) An OSI standard protocol stack, beginning with a link access procedure for the D channel (LAPD) entity.

   3) Any other protocol stack, whether it be standard or proprietary, used for communication in an isochronous channel.

   Coordination of parallel information flows through these stacks shall be a management requirement.

e) In the ISLAN, each ISTE shall coordinate with its AU in their use of MAC services so that the management of each side of the interface is aware of whether the other side is ignoring grant indications in its transmission of MAC frames. This coordination involves peer-to-peer communication by the MAC layer management entities (LMEs) of the ISTE and the AU, using the connectionless service of the MAC.

f) The ISLAN between the AU and its connected ISTEs does not use physical broadcast of MAC protocol data unit (MPDU) frames, in the way, for instance, that a CSMA/CD bus transmission system does. Since the AU reads the MAC service access point (MSAP) in each incoming MPDU to permit routing of the frame to the correct exit port on the AU, the AU can elect to address filter the relayed frames and deliver them only to another attached ISTE or via a bridge to a backbone network. Thus, one option is that local ISTE management may elect to treat as an error the receipt by a particular ISTE of an MPDU frame not addressed to it.

   This feature of address filtering has the desirable consequence of avoiding one of the security concerns about IEEE 802 LANs, i.e., their use of promiscuous receptions, leaving it to the receiving station to decide whether a frame is addressed to it and should, therefore, be processed.

   Alternatively in some configurations, more conventional IEEE 802 policies in which each LAN attachment filters incoming frames, based upon its MAC address and the frames' destination addresses can be useful as part of this filtering process. If, for example, the IEEE 802.9 LANs are configured as a set of hierarchical subnetworks "root," AU ports may be treated as connections to virtual ISTEs, although they are actually IS-AU to IS-AU connections. In this case, there will be a clear need for a "sibling" IS-AU to accept MPDUs with several multiple addresses each for further relaying to their ultimate ISTE destination.

   There is therefore, a MAC LME that provides the facility to optionally report to layer management the receipt of an MPDU frame not addressed to a particular MAC. The default state of the option is that all valid MAC frames are accepted without reporting incorrectly addressed MPDU frames to layer management.

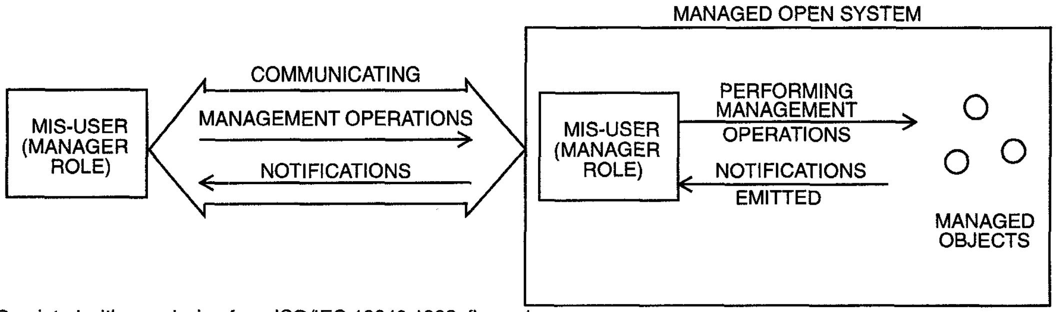
10.3 OSI network management framework

10.3.1 Distributed OSI network management

In the OSI management framework, an MO is conceived to be an abstraction from a real resource, such as a protocol layer entity implementation, an active virtual circuit, or a piece of physical communication equipment (e.g., a LAN interface adapter in a workstation). An MO is thought of as contained in the open system to which the real resource represented by the MO belongs.

The MO is, in effect, an abstract representation of a real resource. As an abstract representation, the MO is an aggregation of information and functions, which can be created and destroyed by a managing process. However, the real resource represented by the MO is not itself created or destroyed by these management operations. Nevertheless, the information elements of the MO are taken directly from the real resource, and the functions directly affect the real resource. Otherwise, the managing process' interactions with an MO would not have any relationship with, and thus could not manage, the corresponding real resource.

In OSI network systems management, MOs are subject to management operations originated from a systems management application process (SMAP), and the MOs, in turn, send management notifications to the SMAP. The managing SMAP may be local to the open system in which MOs have been abstractly defined from real resources, or remotely located across a network from the open system containing the MOs. The SMAP that manages objects is referred to as a "managing process." If the managing process is not local to the open system of an MO, that process cannot directly interact with the MO because of the separation of the open systems across a network. Therefore, a remote managing process shall depend upon a local agent SMAP to directly apply operations to and receive notifications from an MO. In that case, the managing process and the agent process shall exchange messages containing the content of the operations and the notifications, across a network, using the application layer specifications of the common management information services and protocols (CMIS/CMIP). See figure 10-1.



Reprinted with permission from ISO/IEC 10040:1992, figure 1.

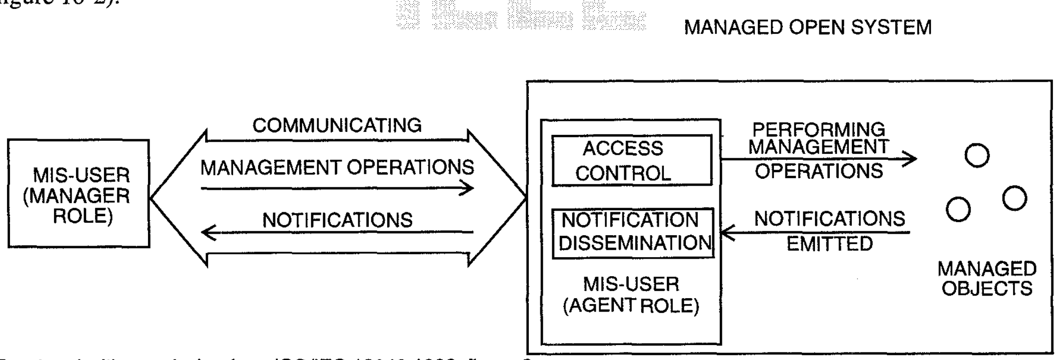
Figure 10-1—Systems management interactions

Examples of management operations performed by an agent SMAP, on behalf of a managing SMAP, are the retrieval of the value of an error counter which is the attribute of a layer entity MO or the initiation of a reset of a PHY entity MO. Examples of notifications sent from an MO, via the agent SMAP, would be a notice that a threshold on a counter of corrupted packets has been exceeded or a notice that a PHY diagnostic self-test has failed.

The managing SMAP and the agent SMAP are users of the management information services (MIS) which provide specific management information at MO boundaries. The agent SMAP processes management information in order to serve the application purposes of the managing SMAP. The latter uses specific management information elements and facilities to perform functions in the five OSI management functional areas, which are fault, configuration, accounting, performance, and security.

The agent SMAP is conceived of as interacting with a set of MOs in a managed open system. The agent SMAP initiates management operations on these local MOs in response to requests from the managing SMAP sent via CMIP protocols. The agent SMAP also receives notifications from these local MOs, which may be forwarded using CMIP protocols to one or more managing SMAPs. However, the interface or other mechanisms, which are used locally by the agent SMAP to interact with its MOs, are beyond the limits of OSI systems management standards.

The agent SMAP performs two additional functions in its role as mediator between the managing SMAP and the MOs in the agent's local open system: access control (AC) and notification dissemination (see figure 10-2).



Reprinted with permission from ISO/IEC 10040:1992, figure 3.

Figure 10-2—Access control and notification discrimination

The agent SMAP is not a dependent entity that serves all managing SMAPs regardless of identity. The agent can deny access by certain managing SMAPs to the information elements and functions of specific MOs. The AC function of the agent SMAP is specified in ISO/IEC 10164.

The agent SMAP also performs the function of sending all and only those notifications generated asynchronously by the MOs in the managed open system. The way in which the discrimination of notifications is carried out so that they are routed to the appropriate managing SMAP is specified in ISO/IEC 10164-5:1993.

The agent SMAP performs a variety of other functions as well, but the only additional function to be commented upon here is that of building and maintaining a log, which is an MO consisting of records of operations on an MO, state changes of an MO, and event notifications received from an MO. The specification of logging such records is given in ISO/IEC 10164-6:1993.

10.3.2 The management information base

The collection of MOs in an open system is conceptually aggregated into a management information base (MIB), which is the persistent object store for the SMAPs, both agent and managing. The MIB contains two types of MOs: layer MOs and systems MOs (see figure 10-3).

There is a set of MOs in the MIB corresponding to each OSI Layer. These objects are related by a containment relationship; for instance, a layer entity MO may contain an SAP MO for that layer or a connection MO controlled by that layer. In addition, there may be a set of systems MOs in the MIB that differ from the layer MOs because the systems MOs are relevant to more than one layer. For instance, there may be a systems MO for the whole station containing the OSI protocol stack. Systems MOs may also be related in a containment hierarchy.

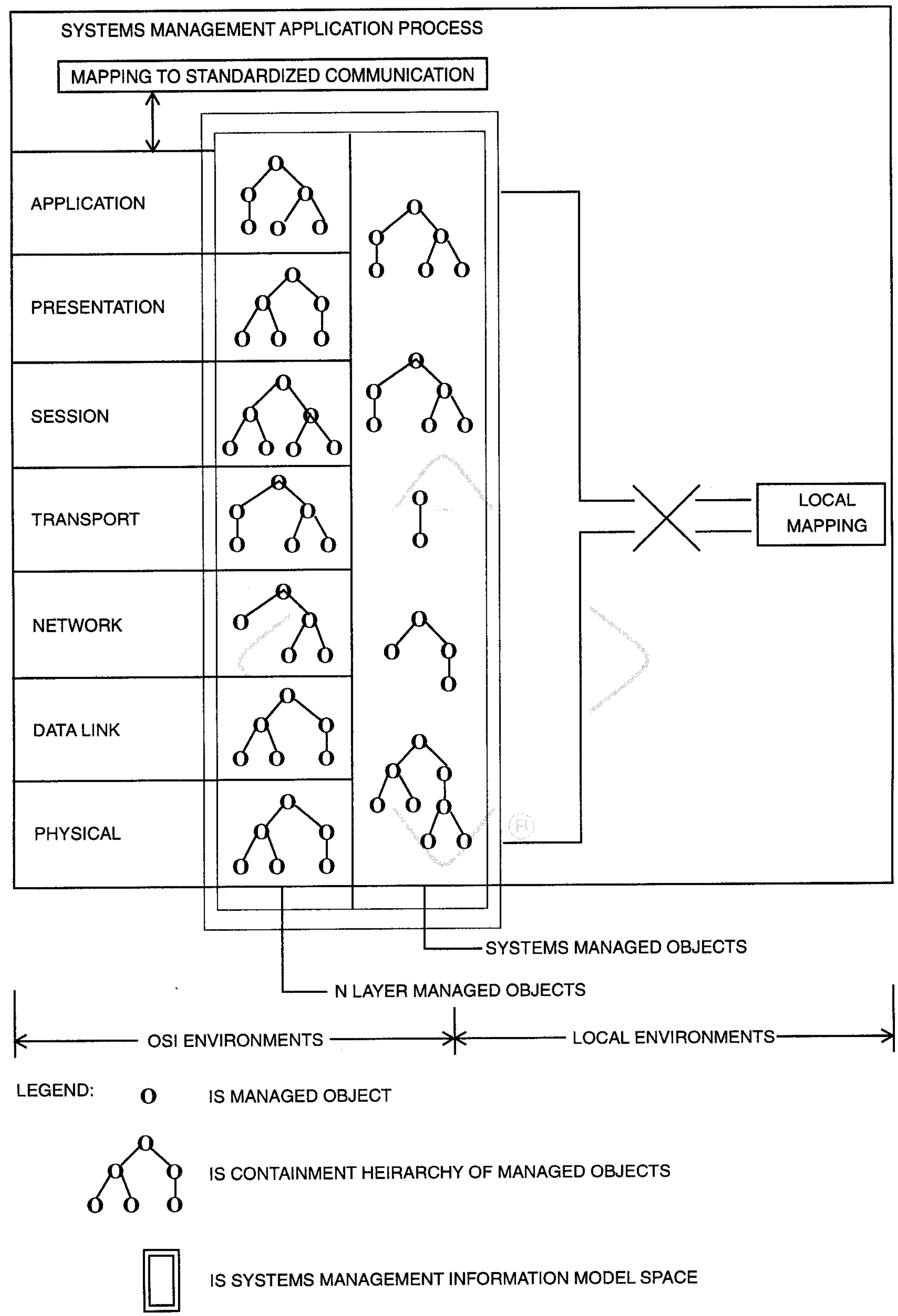
A managing process can request information from an agent process concerning one or more of the MOs in the MIB by exchanging management application protocol data units (MAPDUs), encapsulated within the CMIP data units (CMIPDUs). See figure 10-4.

This exchange of PDUs is governed by an Application Layer protocol. CMIPDUs are exchanged over associations, which are mapped by the association control service element (ACSE), via the Presentation Layer connection environment, onto session (and transport) connections for end-to-end reliable delivery. Remote operations initiated by a remote managing process are communicated to the local agent process using the command association services of the remote operations service element (ROSE).

The OSI systems management framework, as presented in figure 10-4, assumes that CMIS/CMIP services are provided over a full 7-layer OSI protocol stack. However, the LAN/MAN management service (LMMS) and the LAN/MAN management protocol (LMMP), as defined in ISO/IEC DIS 15802-2, provide a more efficient approach in which CMIPDUs can be exchanged between managing and agent SMAPs over a connectionless LLC service.

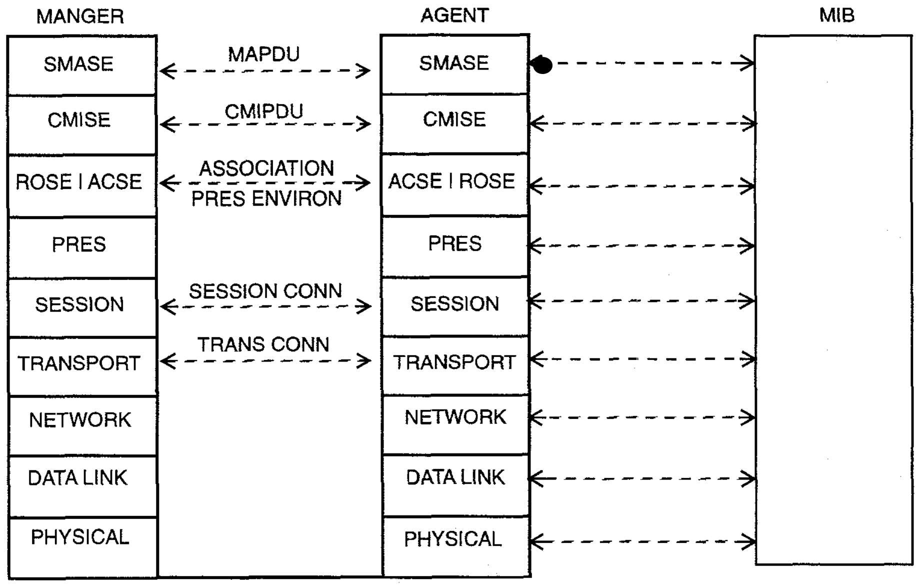
10.3.3 The function of the managing and agent SMAPs in the ISLAN

In the ISLAN, the typical management configuration will involve the AU hosting one or more managing SMAPs and the ISTEs hosting the agent SMAPs. This arrangement is certainly not a requirement, but is attractive for the reason given in 10.2, item b). The protocol, management, and signalling architectures required in the ISLAN favor the building of the AU as a hub for centralized OSI management and CCITT Q.93x signalling (see Chapter 11).



Reprinted with permission from ISO/IEC 10040:1992, figure 2.

Figure 10-3—The management information base



SYMBOLS:

ACSE Association Control Service Element
CMIPDU Common Management Information Protocol Data Unit
CMISE Common Management Information Service Element
MIB Management Information Base
MAPDU Management Application Protocol Data Unit
ROSE Remote Operation Service Element
SMASE Systems Management Application Service Element

Figure 10-4—Exchange of MAPDUs between management information service users

10.3.4 The relationship between OSI systems management and layer management

The OSI systems management framework defines a distributed management architecture for open systems. A local SMAP uses LMEs to monitor and interact with the attributes of the layer entities, which belong to them as MOs. A centralized managing SMAP accesses information on both the layer MOs and the systems MOs in a remote open system by exchange of operations requests and notifications with the remote agent.

The layer management interface (LMI) of each LME provides access to the attributes and methods of specific layer MOs. The LMI may be thought of as an abstraction presenting only certain, management-relevant features of the layer entities.

The LMEs may be conceived of as a set of tools for access to the abstract layer entity MO.

10.4 Relationship of IEEE 802.9 subnetwork management entities

Figure 10-5 illustrates the management entities that participate in a comprehensive and complete system management operation across a subnetwork consisting of the IEEE 802.9 ISTE, IEEE 802.9 AU, and existing backbone LAN and ISDN networks.

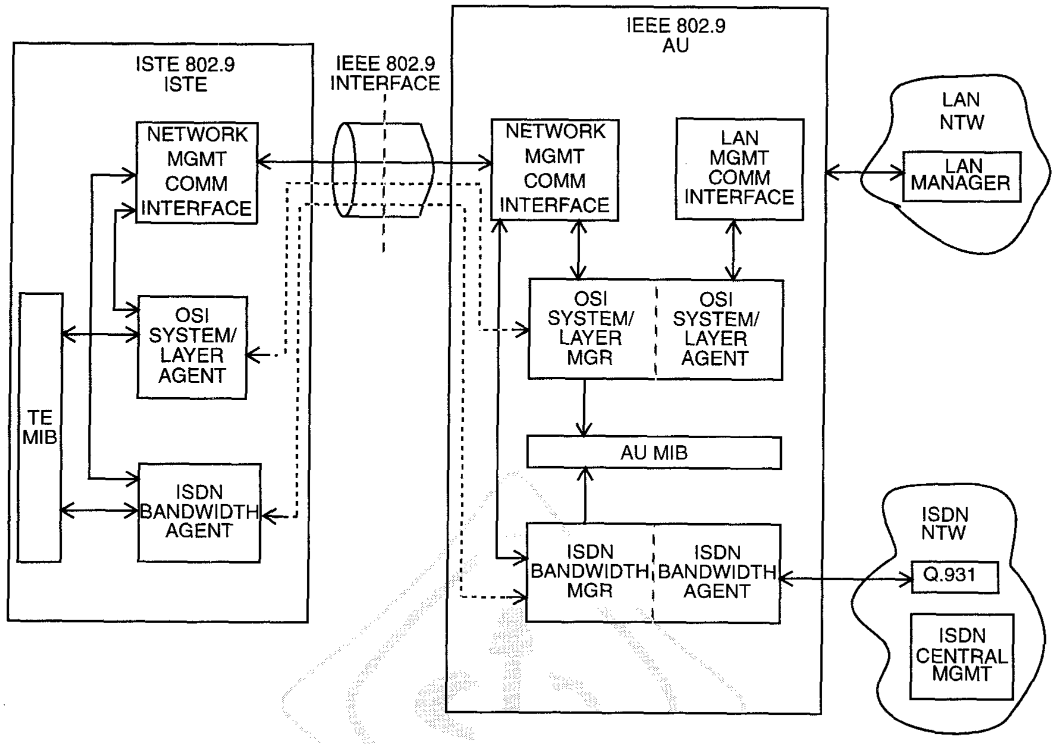


Figure 10-5—Relationship of IEEE 802.9 subnetwork management entities

The IEEE 802.9 ISTE contains four management functional modules:

— Managed information base (MIB)
— Network management communication interface
— OSI system/layer management agent
— ISDN bandwidth management agent

The ISTE has been modeled with two separate management agents for the reason that the OSI system/layer manager is separate from the ISDN network manager under the present deployment configurations. Since the OSI (LAN) network management system is separate from the ISDN wide area network (WAN) management system, there will be two separate agents. Each agent has the ability to independently perform set/get procedures on the MIB within the ISTE. In order to guarantee correct access of control information and status, there must be some interlock coordination between these two ISTE management agents as they access the ISTE's MIB.

The ISTE and the AU each has a communication interface entity for actual exchange of information across the IEEE 802.9 access interface. The network management communication interface is responsible for the presentation of management information to a given channel for transport across the local 802.9 interface. The AU has two separate communication interfaces for the communication with the LAN network manager and the ISDN network manager. The OSI system/layer manager entity has a dual responsibility:

a) It serves as an agent for communication with the backbone network manager.
b) It serves as a manager for the communication with the ISTE.

The three internal management coordination interfaces within the IEEE 802.9 AU are as follows:

— Coordination role between the OSI system/layer agent and the OSI system/layer manager
— Coordination role between the ISDN bandwidth agent and the ISDN bandwidth manager
— Coordination between the ISDN management entity and the OSI system/layer management entity as they interface to the AU MIB

These three internal interfaces are outside the scope of this standard, and are a matter for further study.

10.5 ISDN network management information exchange

The ISDN network management entities in the ISDN WAN, the AU, and the ISTE use the CCITT Q.93x/Q.94x protocol suites for the purpose of message exchange. Figure 10-6 illustrates the exchange of management information in an IEEE 802.9 subnetwork. This figure illustrates multiple CCITT Q.93x signalling streams as follows:

a) The dashed line illustrates the CCITT Q.93x flow from the AU to the ISTE, wherein CCITT Q.93x Codeset 7 is used as a private network signalling protocol along with CCITT Q.93x standard Codeset 0 messages to provide for the negotiation of channel assignment between the AU and the ISTE. (See 11.4.4.1 for a complete description of the codeset usage philosophy.)
b) The solid line between the ISDN WAN and the AU represents the standard Codeset 0 message flow.
c) The dotted line illustrates the CCITT Q.93x flow from the ISDN WAN backbone to the ISTE ISDN application, for the use of channel assignment for ISDN service. In particular, private codesets such as Codeset 6 information may be transparently conveyed to the ISTE in order to support supplementary network management procedures.

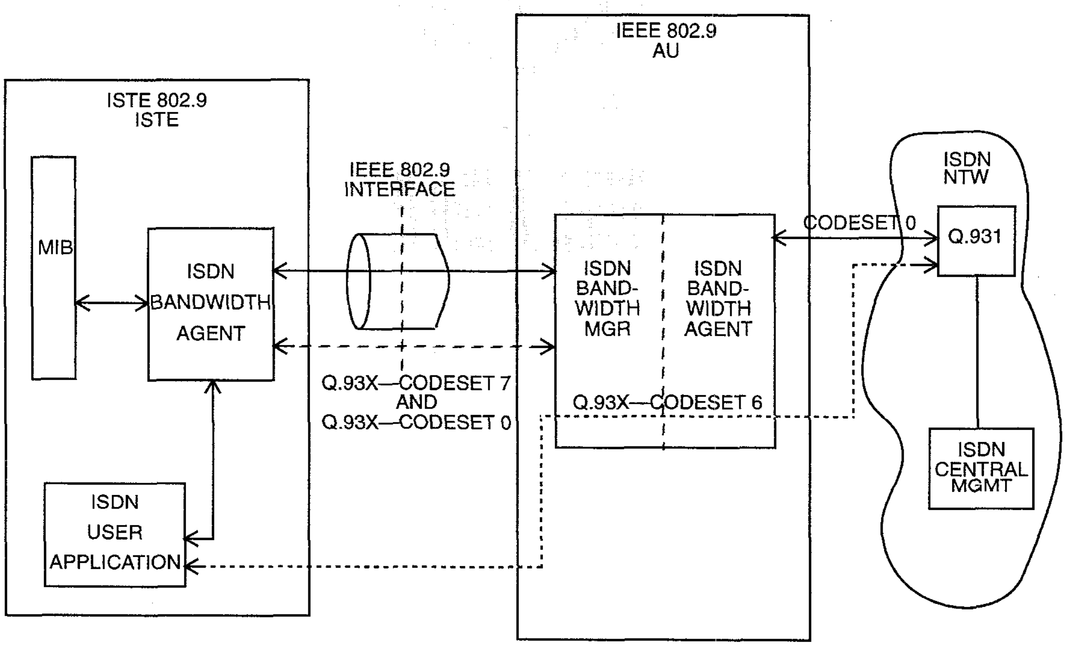


Figure 10-6—ISDN network management information exchange

10.6 Management transport channels within the IEEE 802.9 interface

Figure 10-7 illustrates the transport of the OSI layer management information and the ISDN layer management information across the IEEE 802.9 interface.

The OSI layer management information is communicated between the AU and the ISTE as part of the MAC-based services over the P channel. Note that the specific MAC address to be used for OSI layer management exchange is a matter for further study.

The ISDN layer management information is communicated between the AU and the ISTE as part of the CCITT Q.93x/Q.94x services over the D channel. Specifically, this information is conveyed over the LAPD logical channel of SAPI="63." This would be Codeset 0 information.

The IEEE 802.9 dynamic bandwidth management information is communicated between the AU and the ISTE as part of the CCITT Q.93x services over the D channel. Specifically, this information is conveyed over the LAPD logical channel of SAPI="0". This would be Codeset 7 information.

The integration of the OSI and ISDN layer management information streams over a single channel is a matter for further study. There are two possible configurations:

a) Shared use of the D channel wherein the OSI layer management information is sent over a distinct CCITT Q.93x/Q.94x management message.

b) Shared use of the P channel wherein the OSI layer management information would use the reserved LAN MAC address (SID=01010101), while the ISDN layer management information would be sent a separate convergence specifier (SID=01010110). This would be referred to as the P_D channel.

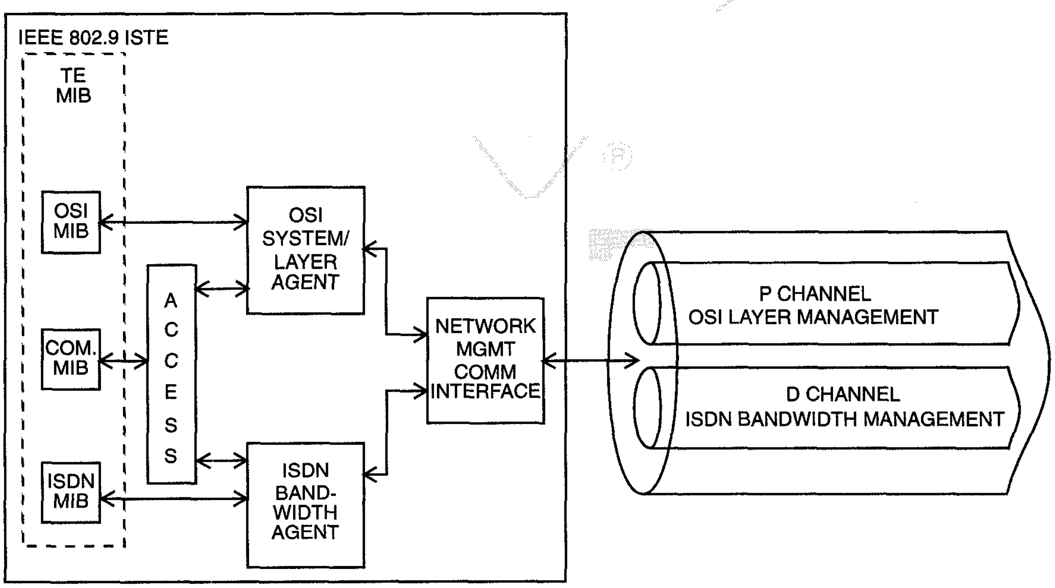


Figure 10-7—Detailed view—Management transport channels within the 802.9 interface

10.7 The structure of OSI management information

The standard for the definition of the data type structure of MOs, specified in ISO/IEC 10165, relies heavily on the theory of object-oriented definition. This object-orientation derives from the theory of object-oriented design and programming. However, the OSI structure of management information (SMI) makes use of no specific object-oriented programming or design language.

SMI is fundamental to management application development since the MO interface provides the information and control that the management application program shall exercise over OSI entities, defined as MOs.

10.7.1 Object-oriented concepts of the structure of MOs

This subclause presents the object-oriented concepts of OSI systems management as a basis for understanding the specifications of the MOs at the ISLAN interface. The concept of the MO is crucial to the three types of management protocols identified in the OSI management framework: systems management, layer management, and N Layer protocols. The definition of the MO is provided in ISO/IEC 10165.

The abstract definitions of classes of MOs define the abstract data types of the objects in the MIB for the use of various management applications. Layer management specifications are concerned with the subset of the MIB that constitutes the layer information base required for layer management. Systems management is concerned with the entire MIB of an open system.

10.7.1.1 Abstraction

An OSI MO is an abstraction of a real entity, e.g., a MAC entity implemented in a real open system. The abstraction is a selection of the characteristics of the real resource to which the MO corresponds. The selection is usually much smaller than the property set of the real resource, because the selection is made on the basis of the limited information requirements of management applications.

MOs are specified by data type, i.e., by reference to the class to which the object belongs, along with all similar objects. The class is defined in an abstract declaration of a data type. It is important to keep clear the distinction between an abstract class of objects and the particular instances, i.e., objects, belonging to that class.

The abstract specification of the MO, i.e., its class, defines what an SMAP is, whether it is local or remote, and whether it can "see" at the abstract boundary of the object. Specifically, three types of information are visible at the management boundary of an object:

a) Attributes of the object (e.g., a counter belonging to a MAC entity)
b) Methods for the manipulation of the object, which are typically read and write operations on the attributes of an object (e.g., an operation to get the present value of a MAC counter)
c) Notifications from the MO of various events of management significance (e.g., a notice that a MAC counter has reached a threshold)

Thus MOs are limited views of particular real resources. These views may be defined and may be represented in the MIB even if the real resources do not exist. However, in that case, the MO ceases to have any significance for management applications.

10.7.1.2 Encapsulation

Data hiding, in the abstract conception of an MO, is achieved through the specification, in the definition of a class, of methods and attributes that are all and only what is exported to client objects. Objects belonging to client classes contain an object from or a reference to an object in the exporting class. The client object can only access those attributes and methods of the contained object that have been exported. All other information content of the contained object is hidden. If a MAC MO provides methods for reading counters of MPDUs that have been rejected because of a failed cyclic redundancy check (CRC) test, counters of the number of MPDUs correctly received, etc., other information about the functioning of the MAC (e.g., the size of the transmit and receive buffers) remains hidden from the agent and the managing SMAPs, which are clients of the MOs.

Encapsulation is the mechanism that ensures that there is no access by objects of a client class to contained objects and their attributes, other than through exported methods.

The attributes of an object are not directly accessible. Their manipulation requires the sending of messages to the object to which they belong. The object receiving the messages is the actual agent of the operation on its attributes. The ways of sending these messages are called the methods of the object. An object's defined methods are the only access to the object and its attributes.

10.7.1.3 Filters

A filter is a Boolean test on the value of an attribute of an MO that determines whether an operation may be performed on that MO. For instance, an integer valued attribute of an MO might be reset to a default value, e.g., 0, if its current value is negative.

IF attribute < 0 THEN set attribute to default 0.

As this example suggests, the filter is specified as an attribute assertion; in this case, attribute < 0.

10.7.1.4 Notifications

Notifications of events in one object are messages sent to another object, which acts as a manager of the object that emits the notifications. Notifications are sent on the satisfaction of conditions that are defined algorithmically in the definitions of appropriate methods of an object.

A PHY MO, for instance, might send the agent SMAP (for relay to the MO in a CMIP message structure) a notification that a threshold on a counter of instances of loss of carrier has been exceeded.

10.7.1.5 Inheritance

Object-oriented specification defines abstract data structures or classes, and specifically (in the OSI management framework) classes of MOs, through the use of the logical technique of refinement. Through refinement, more general classes of objects are logically related to more specific classes by inheritance. The relationship of inheritance is structured as a graph structure (typically a tree) of classes, in which each node MO class is connected to one or more immediate parent MO classes.

The specification of classes must locate them in an inheritance directed graph. A class inherits all of the attributes and methods (i.e., routines) from its immediately superior parent(s) and more ancestral classes. This is called "strict inheritance." In the descent of strict inheritance from parent(s) to child, additional attributes and methods are specified that distinguish child classes from their parent(s) and more remote ancestors.

The definition of a MAC MO class might be a refinement from a parent MO class, layer entity, which might in turn be the child of the ultimate MO class, Top. From its immediate parent, layer entity, such a MAC MO class might inherit the attributes of a service boundary by which it provides communication services for SDUs from the LLC, a counter of PDUs received, etc. Layer entity would inherit from Top, among other attributes, an attribute containing the name of instances of the class. On the other hand, the definition of the MAC MO class would add attributes not found in the ancestor classes, e.g., a counter of PDUs rejected because of a failed CRC test.

Inheritance in the classes of the SMI is usually single (i.e., no descendant class has more than one immediate parent); however, there may be cases where multiple inheritance will be conceptually required.

10.7.1.6 Containment

As noted above, a class the member objects of which contain an object belonging to another class, or which contain a reference to an object in another class, is said to be a "client" of the latter class. This containment relationship allows the construction of a containment tree of classes of objects that contain objects from other classes. This tree may be isomorphic with the composition of the attributes of real entities (e.g., implementations of an ISTE containing a PHY entity, a MAC entity, an LLC entity, etc.).

10.7.1.7 Allomorphism and polymorphism

Allomorphism is a feature of the type compatibility rules of the SMI theory of object-oriented design by which one class is compatible with another, meaning that the objects of the first class can be managed as though they were objects of the second class. Polymorphism is a special form of allomorphism. Polymorphism is a feature of the type compatibility rules of an object-oriented notation by which any descendent class is type compatible with any of its ancestors. This means that any descendent object may be the polymorphic value of a variable reference to an object in one of its ancestral classes. Allomorphism does not require that the compatible classes be in an ancestor-descendant relationship.

Polymorphism is the form in which allomorphism will be used in the specification of the MOs of the ISLAN.

Although it may not be immediately apparent, polymorphism is the most fundamental feature of the theory of object-oriented definition. By virtue of its incorporation in a system of notation, the methods of an ancestral class, executed by its clients, may actually be routines defined in a descendant class. This allows for a powerful way to define general (ancestral) algorithms, and refine incrementally the actual methods executed depending upon the descendant object assigned to an ancestral class variable. Because of polymorphism, the process of refining classes from ancestral classes does not require any corresponding changes in the ancestral methods and the methods of other classes containing variable references to ancestral objects.

Using a pseudo-code resembling C++, the fundamental mechanism of polymorphism can be expressed as follows:

```
class Ancestor_Type {public: virtual void operation( int ) ; };

Ancestor_Type::operation( int ) { ... };

class Descendant_Type : public Ancestor_Type
       { public: void operation( int ); };

Descendant_Type::operation( int ) { ... };

Ancestor_Type object,ancestor_object;
Descendant_Type descendant_object;
```

```
int i = some_constant_value;
if ( boolean_test )
       object = descendant_object
else object = ancestor_type;

object.operation ( i );
```

As shown, Descendant_Type is a class that inherits from the class, Ancestor_Type, and operation is declared to be a virtual method in Ancestor_Type. The characterization "virtual" means that the method can be redefined in descendant types. In the definition of Descendant_Type, operation is redefined, e.g., to take advantage of some optimization made possible by the attributes of the descendant.

Whether the operation is applied to an ancestor_object or a descendant_object, depends upon boolean_test. The actual routine executed by object.operation(i) depends upon whether object is of ancestor_type or descendant_type. If boolean_test is satisfied, then the actual method used when object.operation(i) is executed will be the redefinition of operation for the Descendant_Type. Otherwise, it will be the original method as specified in the definition in Ancestor_Type. Thus, polymorphism achieves generalization through the definition and redefinition of virtual methods.

The importance of allomorphic generalization to OSI MO specifications is that through class refinement new protocols may be defined in the descendant classes of older protocol MO classes. However, the specification of routines [comparable to object.operation(i)] in management applications may remain the same, although covering an ever growing number of protocols. This provides a powerful way of developing successive generations of protocols for different MO classes without breaking management applications.

Another important aspect of allomorphism is that it makes it possible, for instance, for the same SMAPs to manage IEEE 802.3 or IEEE 802.5 protocol objects, as well as IEEE 802.9 protocol objects. The introduction of IEEE 802.9 systems into customer premises installations will often involve the construction of an ISLAN for interworking with other existing IEEE 802 LANs. The feasibility of this integration of different LAN technologies will depend in part on the provision of integrated management of all LANs using existing management applications.

10.7.2 Services for the systems management application process (SMAP)

Abstract templates are defined in ISO/IEC 10165-4:1992 as formal meta-linguistic productions that specify how MOs shall be defined in layer management specifications.

However, the template may also be viewed as a specification of a generic service interface for access by SMAPs to all MOs in the MIB, including the managed layer entity itself. This MO service interface is the LMI.

The key aspects of the syntax and semantics of the templates used in this standard are presented below in this chapter, in order to define the information elements contained in the templates as service access points (SAPs) for SMAPs. More specifically, the templates for the MO class and the package are defined here. For further details on these and other templates, see ISO/IEC 10165-4:1992.

10.7.2.1 The MANAGED OBJECT CLASS template

```
<class-label> MANAGED OBJECT CLASS

      [ DERIVED FROM<class-label>[,<class-label>]*; ]
      [ CHARACTERIZED BY<package-label>[,<package-label>]*; ]
      [ CONDITIONAL PACKAGES<package-label>
       PRESENT IF condition-definition
```

```
[,<package-label>
 PRESENT IF condition-definition]*;
]

REGISTERED AS object-identifier;
supporting productions
condition-definition -> delimited-string
```

The MANAGED OBJECT CLASS template provides an information service to SMAPs by specifying four types of information about the MO class:

a) The parent MO class (introduced by "DERIVED FROM") of the MO class being defined. All MO classes have parents, with the exception of Top, the ultimate MO class in the inheritance hierarchy (see 10.9). It is essential for the SMAP that will use this information service to know the parent MO class in order to know whether the child class has attributes and methods that are not specified in the current definition. These unspecified attributes and methods would belong either to the parent class or some more remote ancestor class of the parent and child.

b) The set (introduced by "CHARACTERIZED BY") of one or more packages of attributes and methods (i.e., operations on attributes, actions, and notifications) that must be present in members of the MO class. As the meta-linguistic syntax of the production indicates, the specification of this set or list of sets is itself optional.

c) The set (introduced by "CONDITIONAL PACKAGES") of one or more packages of attributes and methods that are present in members of the MO class, only if a Boolean condition is satisfied. As the meta-linguistic syntax of the production indicates, the specification of this set or list of sets is itself optional.

d) The ASN.1 registered object identifier (introduced by "REGISTERED AS") for the MO classes defined by this production. This integer vector uniquely identifies the data type of the objects in the class being defined. Combined with a name binding, this mechanism allows for the unique identification of all MOs. The delimited string supporting production is used here, but is also used in other templates. For the purpose of the MO specifications in this standard, delimited strings shall be delimited by "character pairs" at the beginning and end of the string.

10.7.2.2 The PACKAGE template

```
<package-label> PACKAGE

      [ BEHAVIOR<behavior-definition-label>
      [ ,<behavior-definition-label> ]*; n ]

      [ ATTRIBUTES<attribute-label> propertylist [<parameter-label>]*
      [ ,<attribute-label> propertylist
      [ <parameter-label> ]* ]*; ]

      [ ATTRIBUTE GROUPS<group-label> [ <attribute-label> ]*
      [ ,<group-label> [ <attribute-label> ]* ]*; ]

      [ ACTIONS<action-label> [ <parameter-label> ]*
      [ ,<action-label> [ <parameter-label> ]* ]*; ]

      [ NOTIFICATIONS<notification-label> [ <parameter-label> ]*
      [ ,<notification-label>  [ <parameter-label> ]* ]*; ]
```

```
[ REGISTERED AS object-identifier ];
supporting productions

propertylist ->     [ REPLACE-WITH-DEFAULT ]
                    [ DEFAULT  VALUE      value-specifier ]
                    [ INITIAL  VALUE      value-specifier ]
                    [ PERMITTED VALUES    type-reference ]
                    [ REQUIRED  VALUES    type-reference ]
                    [get-replace ]
                    [ add-remove ]

value-specifier->   value-reference    |
                    DERIVATION RULE <behavior-definition-label>
get-replace         GET | REPLACE | GET-REPLACE
add-remove          ADD | REMOVE | ADD-REMOVE
```

The PACKAGE template provides an information service to SMAPs by specifying six types of information about the unconditional and conditional package components of an MO class specification:

a) An optional description (introduced by "BEHAVIOR") of the behavior of the information element constituents of the package. Behavior here means a specification of a variety of detailed considerations about the constituents, which are relevant for management application purposes.

   These might include, among other things, the following:

   1) The semantics of the constituents
   2) The state-specific circumstances in which notifications are produced by the MO
   3) Dependencies between the values of various attributes
   4) Consistency constraints on attributes
   5) Synchronization properties of the MO

   For further description of the concept of behavior in this OSI management context, see Section 5.1.2.4 of ISO/IEC 10165-1:1993.

b) An optional specification (introduced by "ATTRIBUTES") of the names and properties of the individual attributes of the MO.

c) An optional specification (introduced by "ATTRIBUTE GROUPS") of the names and constituents of groups of individual attributes.

d) An optional specification (introduced by "ACTIONS") of actions that an MO, which instantiates the package, may be requested to perform on itself by an agent SMAP. Where required, parameters are also specified for the actions.

e) An optional specification (introduced by "NOTIFICATIONS") of the notifications that an MO, which instantiates the package, can deliver to an agent SMAP. Where required, parameters are also specified for the notifications.

f) An optional specification of the ASN.1 registered object identifier (introduced by "REGISTERED AS") for the package defined by this syntax. Detailed information about the semantics of the PACKAGE template can be found in Section 9.4.3 of ISO/IEC 10165-4:1992.

10.8 OSI management functional areas in the ISLAN environment

The OSI systems management framework defines five functional areas, one or more of which a specific systems management application might address. The current functional areas defined in the framework are as follows:

a) Fault management
b) Accounting management
c) Configuration management
d) Performance management
e) Security management

The following subclauses define the special systems management functions and issues that arise in the ISLAN context, regarding each of the five functional areas. Wherever such functions are directly tied to the LMI of the ISLAN MAC or PHY, this is noted.

10.8.1 Fault management in the ISLAN

OSI fault management is concerned with the detection, diagnosis, and correction of faults, where a fault is any event in an open system that results in abnormal operation. Specific functions include maintenance and examination of error logs, actions upon receipt of a notification of an error, and diagnostic testing.

The range of error events is defined by the systems management applications and can include any OSI protocol-related function, element, or service.

In the ISLAN context, the AU is architecturally well placed to support fault management applications. Thus the AU may conduct connectivity testing to determine whether the point-to-point physical links connecting it to its ISTEs allow both packet communication, within a time-out period, and information flow in isochronous channels. The AU may perform MAC protocol tests to determine whether grant indications transmitted by the AU are being properly observed by the ISTEs. Also PHY Layer loopbacks from the AU to some of the ISTEs can be conducted at the request of a local or remote managing process. The AU itself shall be subject to fault management, either by an ISTE or an open system external to the ISLAN, which hosts a managing SMAP.

10.8.2 Accounting management in the ISLAN

OSI accounting management is concerned with the procedures for determining, logging, and reporting the end-user's or end-system's resource consumption in any given instance of communication. This functional area will be of importance both for organizational managers and end-users. Accounting information might, for instance, form the basis for cost distribution in an organization.

The architecture of LANs, unlike public networks, does not lend itself to the type of operation that depends upon accounting for resource consumption. Use of LAN services would be difficult or impossible to account for precisely, because the IEEE 802 LAN architecture has no requirement for a central processor that can create accounting records.

However, in the ISLAN, access will typically be provided to public ISDN networks and other backbone networks in which services are provided on a charged basis. Therefore, accounting management will usually be most important for the operation of the ISLAN.

Although the accounting of usage of external backbone network services may be handled by an external integrated services private branch exchange (ISPBX), the ISLAN, because its architecture is centered around an intelligent AU, has a natural basis for account management. Of course, there is nothing in the

PHY or MAC architectures of the ISLAN interface that requires that the AU have the resources to maintain accounting data and carry out local or remote accounting functions.

10.8.3 Configuration management in the ISLAN

OSI configuration management is concerned with the creation, definition, naming, and deletion of MOs; the modification of the states of MOs; the modification of the values of the attributes of the MOs; and the creation, modification, and deletion of relationships between MOs.

There are a number of ISLAN-specific configuration management functions:

a) The management of the table in the AU that identifies the ISTEs directly connected to it. The table is modified as TE is added or removed from the ISLAN. The table will be used, among other things, to allow the AU to perform its role of address filtering. This table management is a MAC layer management function, since the TEs will be identified by MSAP addresses.

b) The management of the attribute of the AU MAC and the ISTE MAC that determines whether the transmission peer ignores grant indications when transmitting an MPDU. At initialization or reset, this attribute is set to IGNORE_GRANT=YES for the AU MAC and to IGNORE_GRANT=NO for the ISTE MAC. However, a systems management application can perform the configuration management function of reversing this setting in either the AU MAC or the ISTE MAC.

10.8.4 Performance management in the ISLAN

OSI performance management is concerned with measuring and tuning the performance of communication between open systems. The relevant measures include overloading of communication resources, throughput, waiting times for availability of resources, end-to-end transmission time for a frame, and measures of quality of service (QOS).

ISLAN performance covers both packet communication through a P or D channel and communication through an isochronous channel. So, as examples, not only MPDU frame throughput but also nonblocking access for circuit-switched voice conversations maintained in B channels and video and other transmission through C channels shall be considered in the performance monitoring of the ISLAN.

In the ISLAN context, almost all performance issues are beyond the scope of standardization, since they relate to the internal, implementation-dependent design of the AU. The way in which the AU is provisioned with transmit and receive buffers and internal communication links (e.g., a high-speed bus between line interfaces and buffers) will determine the performance of ISLAN packet communication, as well as nonblocking access for isochronous communication.

10.8.5 Security management in the ISLAN

OSI security management is concerned with the issues of managing security policy, security services (e.g., AC), and security mechanisms (e.g., AC mechanisms) in a network of open systems. These functions are not substantially different in the ISLAN context than those of other LAN architectures, so far as traditional data communication is concerned. However, the ISLAN also provides services for voice, video, facsimile, and other digital information transmission. Security requirements for multimedia applications, and the management of these requirements, remain to be defined.

10.9 Inheritance hierarchy for ISLAN management

The MO classes of the ISLAN form a conceptual inheritance hierarchy that explains the information structures that define what can be managed in this network environment. Figure 10-8 represents the inheritance hierarchy of the systems of the ISLAN.

It is necessary to define the ISTE unit (e.g., an IS workstation) as a managed system so that it can be managed by an SMAP. Similarly, management requirements lead to the definition of the AU system. The iste-system and the au-system MO classes derive by inheritance from the OSI generic system MO class, which in turn derives by inheritance from the supreme MO class, top. The detailed specification of these ISLAN system MO classes is given in 10.13.

In figures 10-8 and 10-9, the standards responsible for the specifications of the MO classes in these hierarchies are named to the side of the box representing the classes.

Figure 10-9 represents the inheritance hierarchy of the ISLAN layer MO classes.

The detailed specifications of the ISLAN layer MO classes will be given in 10.13. The ultimate root of every inheritance tree of MO classes, the MO class Top, contains several important attributes that are inherited by all MO classes, including those in the ISLAN management environment.

```
top  MANAGED OBJECT CLASS

     -- As defined in Subsection 5.15 of ISO/IEC 10165-2:1992, slightly
     -- revised in conformity with ISO/IEC 10165-4:1992.

CHARACTERIZED BY topPackage

REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
     managedObjectClass(3) smi2MObjectClass(15)};
```

10.9.1 The topPackage package definition

```
topPackage  PACKAGE

     BEHAVIOR DEFINITIONS topBehavior;

     ATTRIBUTES

            allomorphs:       GET,
            name:             GET,
            nameBindings:     GET,
            objectClass:      GET,
            packages:         GET;

     REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
            package(4) smi2Package(0) };
```

10.9.2 The topBehavior behavior definition

```
topBehavior  BEHAVIOR

     DEFINED AS

     This is the top level of the Managed Object Class
     Hierarchy, and every other Managed Object Class is a
     specialization of either this generic class (top) or a
     specialization of a subclass of top. *;
```

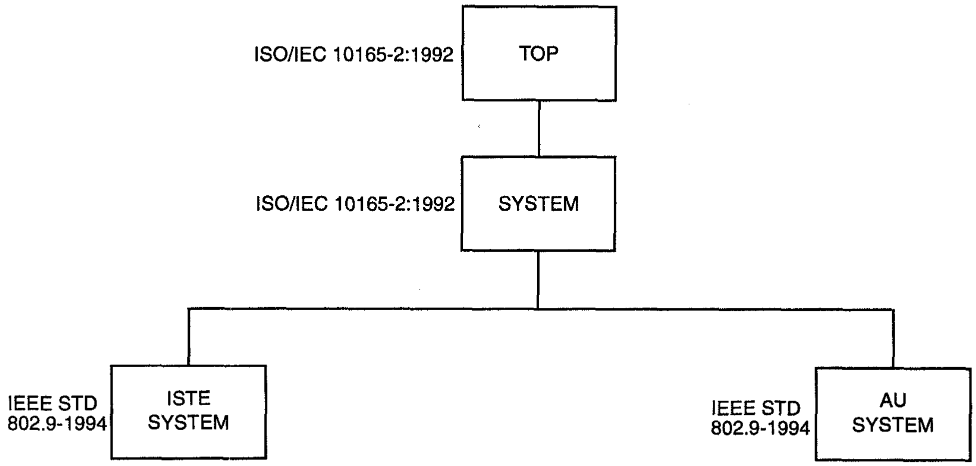


Figure 10-8—Inheritance hierarchy of ISLAN systems

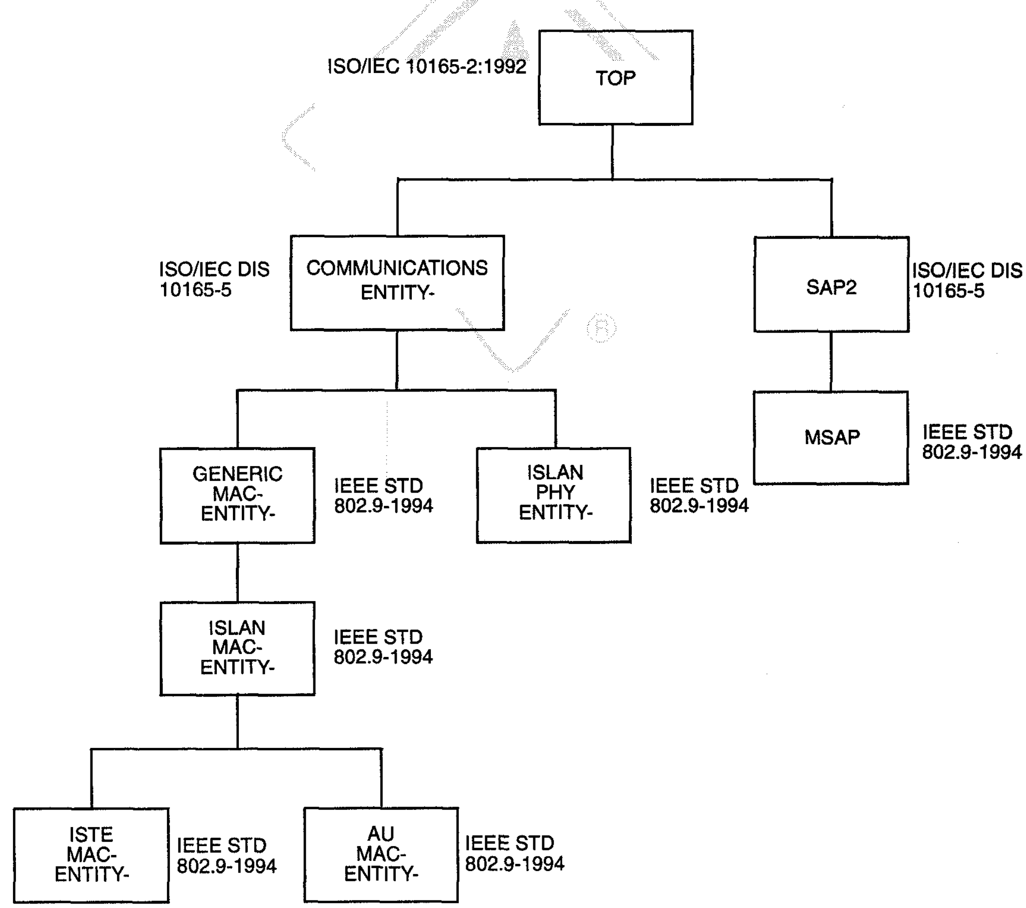


Figure 10-9—The ISLAN MO inheritance hierarchy

10.9.3 Attributes of the top MO

```
allomorphs  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Allomorphs;
      MATCHES FOR Set Comparison, Set Intersection;
      BEHAVIOR allomorphsBehavior;

      REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
             attribute(7) smi2AttributeID(84) };

      allomorphsBehavior  BEHAVIOR

      DEFINED AS

      "The allomorphs attribute appears in every Managed
      Object and indicates the list of allomorphic superclasses
      (if any) supported by the Managed Object instance."
      Subsection 9.7.4.2, ISO/IEC 10165-2:1992. *;

name  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX
             InformationFramework.RelativeDistinguishedName;
             -- Defined in the CCITT Recommendation X.500 OSI Directory
                standards.
      MATCHES FOR Equality;
      BEHAVIOR nameBehavior;

      REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
             attribute(7) smi2AttributeID(97) };

nameBehavior  BEHAVIOR

      DEFINED AS

      "The name attribute type appears in every Managed
      Object and indicates the generic reference to the name of
      a Managed Object relative to a superior Object."
      Subsection 9.7.4.15, ISO/IEC 10165-2:1992. *;

nameBindings  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.NameBindings;
      MATCHES FOR Set Comparison, Set Intersection;
      BEHAVIOR nameBindingsBehavior;

      REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
             attribute(7) smi2AttributeID(98) };

nameBindingsBehavior  BEHAVIOR

      DEFINED AS
```

```
      "The nameBindings attribute type appears in every
      Managed Object and indicates the list of name bindings
      supported by the Managed Object instance." Subsection
      9.7.4.16, ISO/IEC 10165-2:1992. *;

objectClass  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX CMIP-1.ObjectClass;
      MATCHES FOR Equality, Ordering;
      BEHAVIOR objectClassBehavior;

      REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
             attribute(7) smi2AttributeID(100) };

objectClassBehavior  BEHAVIOR

      DEFINED AS

      "The objectClass attribute type appears in every
      Managed Object and indicates the Object class to which the
      Object belongs." Subsection 9.7.4.18, ISO/IEC 10165-2:1992.*;

packages  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.Packages;
      MATCHES FOR Set Comparison, Set Intersection;
      BEHAVIOR packagesBehavior;

      REGISTERED AS { joint-iso-ccitt(2) ms(9) smi(3) part2(2)
             attribute(7) smi2AttributeID(101) };

packagesBehavior  BEHAVIOR

      DEFINED AS

      "The packages attribute type appears in every Managed
      Object and indicates the type of conditional packages (if
      any) supported by the Managed Object." Subsection
      9.7.4.19, ISO/IEC 10165-2:1992. *;
```

10.10 The containment hierarchy for ISLAN management

Figure 10-10 presents, in a tree structure, the containment model for protocol elements of the ISLAN interface.

The ISTE system, as an MO, contains layer entity MOs and their corresponding LMEs. In addition, the ISTE system MO contains a CCITT Q.93x signalling controller MO. The MAC MO contains, as MOs, the processes (SDL symbols) as specified in annex F. The PHY MO is not subdivided into HMUX, PS, and PMD MOs, because these are simply logical packages of management information as far as OSI management is concerned. See the Guidelines for the Definition of Managed Objects (GDMO) specifications of the PHY MO class in 10.2. The specification of the management information elements of the AU system, defined as an MO, is presented in figure 10-11. Although the containment trees are similar,

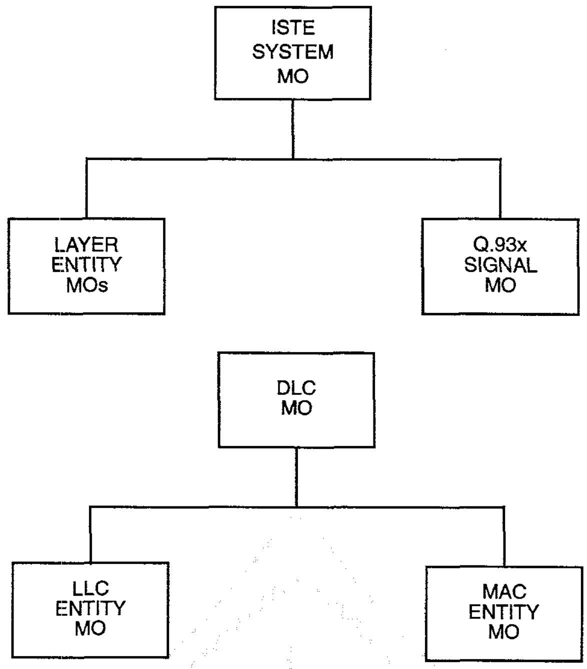


Figure 10-10—The ISTE MO containment tree

the AU system tree structure differs in some respects from that of the ISTE MO. Specifically, the AU line interface does not have layer MOs above Layer 3.

This containment model for the AU system implies that the SMAP(s) in the AU must use the LMMS/LMMP communication service for the exchange of management messages, since there is not a seven-layer protocol stack in the AU to support the CMIS/CMIP exchanges.

10.11 The OSI registration hierarchy for ISLAN layer management

The registration of MO classes and their information elements (e.g., attributes) is required so that an SMAP (either the agent or the manager) can identify the type of resource that is the focus of systems management functions. Registration involves the assignment of an integer vector (i.e., an ASN.1 object identifier) that encodes the hierarchy of authority for the standard designation of MOs and their information elements.

Figure 10-12 presents the registration authority tree, as it is currently understood.

The stub "islan" will be used, in all registration vectors in this text, to represent the highest authorization node that represents the IEEE 802.9 domain.

The second element of the hierarchy description for MOs has an identification plan as follows:

managed-object-class	(3)
package	(4)
parameter	(5)
name binding	(6)
attribute	(7)
action	(9)
notification	(10)

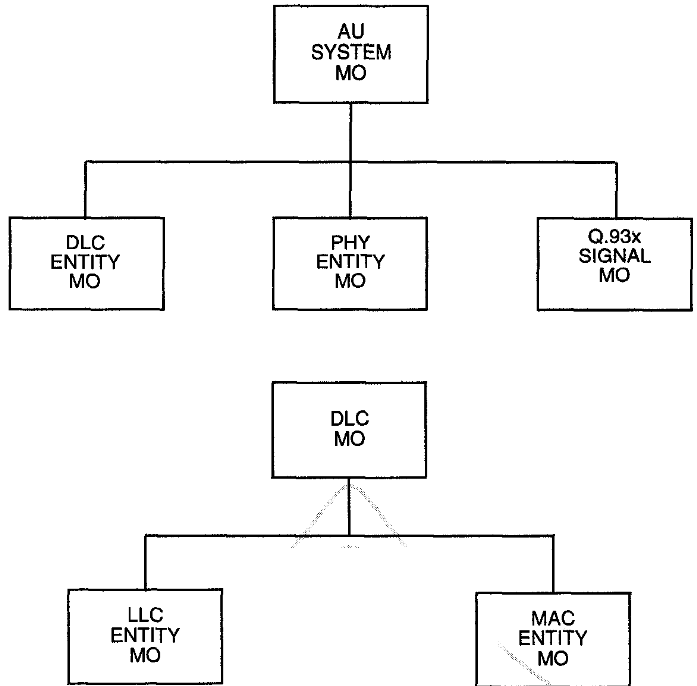


Figure 10-11—The AU line interface MO containment tree

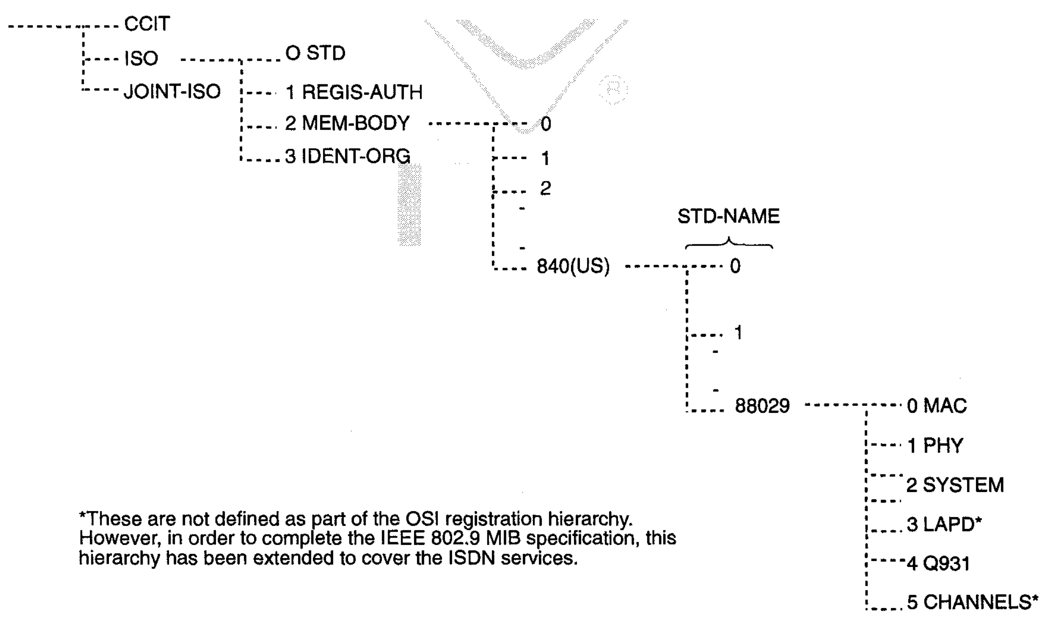


Figure 10-12—The OSI registration hierarchy

10.12 The definition of the ISLAN systems MOs

OSI systems management of the ISLAN concerns itself with the attributes and methods of MOs relevant to the various layer entities implemented in the ISTE and the AU. However, the physically independent systems, represented by the ISTE and the AU, shall also be managed. To provide the information services needed for the management of ISLAN systems, the MO classes for the systems need to be defined.

```
iste-system:  MANAGED OBJECT CLASS
      DERIVED FROM: "ISO/IEC 10165-2:1992": system;
      CHARACTERIZED BY: iste-system-package;

      REGISTERED AS:{ islan, system(2), iste-system(0) };

au-system:  MANAGED OBJECT CLASS
      DERIVED FROM:  "ISO/IEC 10165-2:1992": system;
      CHARACTERIZED BY: au-system-package;

      REGISTERED AS:{ islan, system(2), au-system(1) };
```

The specification of iste-system-package and au-system-package is a matter for further study. These packages will provide additional attributes and methods, beyond those of the system MO class.

Several of the attributes of the system MO class, defined in ISO/IEC 10165-2:1992, are of particular significance in the management of the ISLAN systems:

```
      administrativeState      GET-REPLACE;
      operationalState         GET;
      usageState               GET;

administrativeState  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX
      Attribute-ASN1Module.AdministrativeState;
      MATCHES FOR EQUALITY;

      REGISTERED AS  { smi2AttributeID 66 };

operationalState  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX
      Attribute-ASN1Module.OperationalState;
      MATCHES FOR EQUALITY;

      REGISTERED AS  { smi2AttributeID 71 };

usageState  ATTRIBUTE

      WITH ATTRIBUTE SYNTAX
      Attribute-ASN1Module.UsageState;
      MATCHES FOR EQUALITY;

      REGISTERED AS  { smi2AttributeID 73 };
```

Corresponding to these GDMO are the following ASN.1 specifications (of the attribute syntaxes) of the three attributes:

```
AdministrativeState ::=
ENUMERATED { locked(0),unlocked(1),shuttingDown(2) }

OperationalState ::=
ENUMERATED { disabled(0),enabled(1) }

UsageState ::=
ENUMERATED { idle(0),active(1),busy(2),unknown(3) }
```

These ASN.1 specifications present generic state values characterizing the system as a whole. The detailed description of the generic state management model is defined in ISO/IEC 10165-2:1992. The specific ISLAN system state values require further study.

10.13 Definition of the ISLAN layer MO classes

The MAC and PHY entities and their corresponding LMEs in the ISTE and in the AU are MOs from the abstract standpoint of OSI systems management. These MOs, in turn, contain various constituent MOs.

The following subclauses define these ISLAN MOs, in conformance with the GDMO specified in ISO/IEC 10165-4:1992.

10.13.1 ISLAN MAC MO classes

As indicated above, the MO class specification defines a service interface for a local SMAP (referred to as the agent) or a remote SMAP (referred to as the manager). The following MO class specifications define all of the services provided directly to the local agent, at the ISLAN interface, for the management of the MAC sublayer.

10.13.1.1 MSAP MO specification

The MAC service access point (MSAP) is specified as an MO, following the recommendation of ISO/IEC 10165-4:1992 (Section 9.7). The MSAP MOs are contained within a MAC entity MO.

NOTE—The ASN.1 object identification values, specified in this standard, for various elements of management information are a matter for further study. Wherever possible, arc values have been employed that have already been assigned by an appropriate registration authority.

```
msap  MANAGED OBJECT CLASS

      DERIVED FROM "ISO/IEC DIS 10165-5":sap2;
      CHARACTERIZED BY msap-package;

      REGISTERED AS { islan mac(0) managedObjectClass(3) msap(0) };
```

10.13.1.1.1 msap-package package

```
msap-package  PACKAGE

      BEHAVIOR DEFINITIONS  msap-behavior;
      ATTRIBUTES  mac-address  GET,
            mac-service-user  GET;
```

```
    REGISTERED AS { islan mac(0) package(4) msap-package(0)};
```

The information elements of and relevant to the MSAP MO are fully specified in C.2.

10.13.1.2 Generic MAC entity MO specification

Before specifying the ISLAN MAC entity MO, it is necessary to first define a generic MAC entity MO, corresponding to the generic service specification presented in ISO/IEC 10039:1991. The commonly useful information types specified in the GDMO in IEEE Std 802.1F-1993 have been used to specify some management information elements of the generic MAC MO.

The ISLAN MAC entity MO is a specialization of this generic MAC entity MO, and therefore inherits all of its management characteristics.

```
generic-mac-entity  MANAGED OBJECT CLASS

    DERIVED FROM "ISO/IEC DIS 10165-5": communicationsEntity;
    CHARACTERIZED BY generic-mac-characteristics;

    REGISTERED AS { islan mac(0) managedObjectClass(3)
           generic-mac(1) };
```

10.13.1.2.1 Generic-mac-characteristics package

```
generic-mac-characteristics  PACKAGE

    BEHAVIOR DEFINITIONS  generic-mac-behavior;

    ATTRIBUTES
           mac-entity-id                 GET,
           manufacturer-id               GET,
           manufacturer-product-id       GET,
           manufacturer-product-version  GET,
           max-mpdu-size                 GET
                                         INITIAL VALUE 0
                                         REQUIRED VALUES 5119,

           max-mpdu-size-received        GET
                                         REPLACE WITH DEFAULT
                                         DEFAULT VALUE 0
                                         INITIAL VALUE 0
                                         PERMITTED VALUES
                                         INTEGER (19...5119),

           max-priority-requested        GET
                                         REPLACE WITH DEFAULT
                                         DEFAULT VALUE 0
                                         INITIAL VALUE 0
                                         PERMITTED VALUES
                                         INTEGER (0...7),

           min-mpdu-size                 GET
                                         INITIAL VALUE 19
                                         REQUIRED VALUES 19,
```


Juniper Ex 1032 - 250
Juniper v Chrimar
IPR2016-01399

```
min-priority-requested          GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...7),

mpdu-octets-received-count      GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

mpdu-octets-sent-count          GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

mpdu-received-count             GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

mpdu-received-in-error-count    GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

mpdu-received-in-error-thres    GET-REPLACE
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 255
                                INITIAL VALUE 255
                                PERMITTED VALUES
                                INTEGER (0...1023),

mpdu-sent-count                 GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

priority-last-requested         GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
```

```
                               PERMITTED VALUES
                               INTEGER (0...7),

qos-profile-last-requested     GET
                               PERMITTED VALUES
                               -- For further study

resource-type                  GET,

rx-max-err-count               GET
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 0
                               INITIAL VALUE 0
                               PERMITTED VALUES
                               INTEGER (0...4294967295),

rx-max-err-count-thres         GET-REPLACE
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 255
                               INITIAL VALUE 255
                               PERMITTED VALUES
                               INTEGER (0...1023),

rx-min-err-count               GET
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 0
                               INITIAL VALUE 0
                               PERMITTED VALUES
                               INTEGER (0...4294967295),

rx-min-err-count-thres         GET-REPLACE
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 255
                               INITIAL VALUE 255
                               PERMITTED VALUES
                               INTEGER(0...1023),tab(:); l l.

rx-size-last-mpdu              GET
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 0
                               INITIAL VALUE 0
                               PERMITTED VALUES
                               INTEGER (0...5119),

rx-grant-timer-err-count       GET
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 0
                               INITIAL VALUE 0
                               PERMITTED VALUES
                               INTEGER (0...4294967295),

rx-grant-timer-err-thres       GET
                               REPLACE WITH DEFAULT
                               DEFAULT VALUE 16
```

```
                                        INITIAL VALUE 16
                                        PERMITTED VALUES
                                        INTEGER (0...1023),

        rx-res-err                      GET
                                        REPLACE WITH DEFAULT
                                        DEFAULT VALUE 0
                                        INITIAL VALUE 0

        max-request-time                GET
                                        REPLACE WITH DEFAULT
                                        DEFAULT VALUE 0
                                        INITIAL VALUE 0
                                        PERMITTED VALUES
                                        INTEGER (0...4294967295),

        max-req-err-count               GET
                                        REPLACE WITH DEFAULT
                                        DEFAULT VALUE 0
                                        INITIAL VALUE 0
                                        PERMITTED VALUES
                                        INTEGER (0...4294967295),

        max-req-err-count-thres         GET
                                        REPLACE WITH DEFAULT
                                        DEFAULT VALUE 0
                                        INITIAL VALUE 0
                                        PERMITTED VALUES
                                        INTEGER (0...4294967295),

        standard-revision               GET;

    ATTRIBUTE GROUPS
        -- The following group attribute corresponds to the
        -- ResourceTypeID object defined in IEEE Std 802.1F-1993.

            resource-type-id
            manufacturer-id
            manufacturer-product-id
            manufacturer-product version
            resource-type
            standard-revision;
            --   IEEE   Std   802.1F-1993   also   includes   an   options
attribute
            -- for LME specific optional attributes.

    ACTIONS
        activate-mac        initial-activation-values,
        deactivate-mac,
        reset-mac           initial-reset-values;

    NOTIFICATIONS
        nb-mpdu-received-in-error-thresh-not,
        rx-max-err-count-thres-not,
```

```
            rx-min-err-count-thres-not;
            rx-grant-timer-thres-not;
            max-req-err-count-thres-not;

     REGISTERED AS { islan mac(0) package(4) generic-mac-char(1) };
```

The information elements relevant to the generic MAC entity MO are specified fully in C.2.3.

10.13.1.3 ISLAN MAC entity object specification

The ISLAN MAC MO has two forms: that which represents the MAC in the ISTE and that which represents the MAC in the AU. They differ primarily in the packages of attributes that characterize the different roles of the two transmission peers on each point-to-point link.

```
islan-mac-entity  MANAGED OBJECT CLASS

     DERIVED FROM  generic-mac-entity;
     ALLOMORPHIC SET  generic-mac-entity;
     CHARACTERIZED BY  core-islan-mac-characteristics;
     CONDITIONAL PACKAGES
     iste-mac-characteristics PRESENT IF value( islan-mac-type )=iste *,
     au-mac-characteristics PRESENT IF value( islan-mac-type )=au *;

     REGISTERED AS { islan mac(0) managedObjectClass(3) islan-mac(2) };
```

10.13.1.3.1 Packages of the ISLAN MAC entity MO

The following subclauses define the core and conditional packages of the ISLAN MAC entity MO.

10.13.1.3.1.1 Core ISLAN MAC entity package

The package, core-islan-mac-characteristics, contains the characteristics that belong to all ISLAN MAC entities.

```
core-islan-mac-characteristics  PACKAGE

     BEHAVIOR DEFINITIONS  core-islan-mac-behavior;

     ATTRIBUTES
            islan-mac-type           GET
                                     PERMITTED VALUES
                                     Generic-MAC-Attribute-Module.
                                     IslanMacTypes,

            max-grant-time           GET-REPLACE
                                     REPLACE WITH DEFAULT
                                     DEFAULT VALUE 100 ms.
                                     PERMITTED VALUES for further study.
                                     INITIAL VALUE 100 ms,

            mpdu-to-send             GET
                                     REPLACE WITH DEFAULT
                                     DEFAULT VALUE 0
                                     INITIAL VALUE 0
```

```
                                PERMITTED VALUES
                                BOOLEAN(0-FALSE, 1-TRUE),

        hmux-mac-address-enable GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                BOOLEAN(0-ENABLE, 1-DISABLE),

        rx-grant-time           GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 500 ms
                                INITIAL VALUE 500 ms
                                PERMITTED VALUES
                                INTEGER(0...16383);

    REGISTERED AS { islan mac(0) package(4) core-islan-mac-char(2) };
```

The information elements relevant to the core-islanmac-characteristics package are specified fully in C.2.4.

10.13.1.3.1.2 ISTE MAC specific package

The package, iste-mac-characteristics, contains the characteristics that are distinctive of all ISTE MAC entities.

```
iste-mac-characteristics  PACKAGE

    BEHAVIOR DEFINITIONS  iste-mac-behavior;

    ATTRIBUTES
        iste-type-id            GET,

        iste-manufacturer-id    GET,

        iste-aged-mpdu-discard-count
                                GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...127),

        iste-aged-mpdu-discard-thres
                                GET-REPLACE
                                REPLACE WITH
                                DEFAULT
                                DEFAULT VALUE 127
                                INITIAL VALUE 127
                                PERMITTED VALUES
                                INTEGER (0...1023),
        iste-max-send-mpdu-q-age
                                GET-REPLACE
                                REPLACE WITH DEFAULT
```

```
                                      DEFAULT VALUE 20ms
                                      INITIAL VALUE 20ms
                                      PERMITTED VALUES
                                      INTEGER (0...128ms);

            iste-ignore-grant         GET-REPLACE
                                      REPLACE WITH DEFAULT
                                      DEFAULT VALUE 0
                                      INITIAL VALUE 0
                                      PERMITTED VALUES
                                      BOOLEAN(0-not-ignore-grant,
                                       1-ignore-grant),

      NOTIFICATIONS
            iste-aged-mpdu-discard-not;

      REGISTERED AS { islan mac(0) package(4) iste-mac-char(3) };
```

The information elements relevant to the iste-mac-characteristics package are fully specified in C.2.5.

10.13.1.3.1.3 AU MAC specific package

The package, au-mac-characteristics, contains the characteristics that belong to AUs, by contrast with ISTEs.

```
au-mac-characteristics  PACKAGE

      BEHAVIOR DEFINITIONS      au-mac-behavior;

      ATTRIBUTES
            au-type-id                     GET,

            au-manufacturer-id             GET,

            au-aged-mpdu-discard-count     GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...511),

            au-aged-mpdu-discard-thres     GET-REPLACE
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 511
                                           INITIAL VALUE 511
                                           PERMITTED VALUES
                                           INTEGER (0...1023),

            au-max-send-mpdu-q-age         GET-REPLACE
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 20mS
                                           INITIAL VALUE 20mS
                                           PERMITTED VALUES
                                           INTEGER (0...127mS)
```

```
            au-ignore-grant                 GET-REPLACE
                                            REPLACE WITH DEFAULT
                                            DEFAULT VALUE 1
                                            INITIAL VALUE 1
                                            PERMITTED VALUES
                                            BOOLEAN(0-not-ignore,
                                             1-ignore),

            nb-trans-peers                  GET-REPLACE
                                            REPLACE WITH DEFAULT
                                            DEFAULT VALUE 1
                                            PERMITTED VALUES
                                            INTEGER (0...16),

      NOTIFICATIONS
            au-aged-mpdu-discard-not;

      REGISTERED AS { islan mac(0) package(4) au-mac-char(4) };
```

The information elements relevant to the au-mac-characteristics package are fully specified in C.2.6.

10.13.2 ISTE PHY MO class

The following MO class specifications define all of the services provided directly to the local agent SMAP, at the ISLAN interface, for the management of the PHY Layer sublayers (HMUX, PS, and PMD).

```
islan-phy-entity  MANAGED OBJECT CLASS

      DERIVED FROM "ISO/IEC DIS 10165-5":communicationsEntity;

      CHARACTERIZED BY hmux-characteristics,
            ps-characteristics,
            pmd-characteristics;

      REGISTERED AS { islan phy(1) managedObjectClass(3) islan-phy(0) };
```

The standards for the ISLAN PHY entity are not spread across different standards, by sublayer, as is the case with the data link entity. Therefore, there is no administrative or technical requirement to view the PHY entity MO as containing three sublayer MOs. However, clarity is improved by logically dividing the attributes and methods of the PHY MO class into three packages, as is done above.

The ISLAN PHY entity has been specified in terms of three sublayers: HMUX, PS, and PMD. The following subclauses provide the detailed specifications of these three sublayers.

10.13.2.1 HMUX characteristics package

```
hmux-characteristics  PACKAGE

      BEHAVIOR  hmux-char-behavior;

      ATTRIBUTES
            hmux-type-id          GET,
            manufacturer-id       GET,
```

```
        manufacturer-product-id GET,
        manufacturer-product-version-id
                                GET,

        tx-mode-control         GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0

PERMITTED VALUES
                                INTEGER(0...7),
        rx-mode-status          GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0

PERMITTED VALUES
                                INTEGER (0...7),
        mode                    GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...7),

        mf-error-count          GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...255),

        mf-error-thres          GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 16
                                INITIAL VALUE 16
                                PERMITTED VALUE (16),

        d-chan-rate-control     GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 16 kb/s
                                INITIAL VALUE 16 kb/s
                                PERMITTED VALUES (16 kb/s, 64 kb/s),

        d-chan-rate-status      GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 16 kb/s
                                INITIAL VALUE 16 kb/s
                                PERMITTED VALUES (16 kb/s, 64 kb/s),

        nb-ac-fields            GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 1
                                INITIAL VALUE 1
```

```
                                   PERMITTED VALUES
                                   INTEGER (1...5),

          secure-control           GET-REPLACE
                                   REPLACE-WITH-DEFAULT
                                   DEFAULT VALUE 0
                                   INITIAL VALUE 0
                                   PERMITTED VALUES
                                   BOOLEAN
                                     (0-no security,1-invoke security),

     ACTIONS
          set-d-chan-rate-control;
          set-d-chan-rate-status;

     NOTIFICATIONS
          mf-error-thres-exceeded;

     REGISTERED AS { islan phy(1) package(4) hmux-char(0) };
```

10.13.2.2 PS characteristics package

```
ps-characteristics  PACKAGE

     BEHAVIOR  ps-char-behavior;

     ATTRIBUTES
          rem-loopback-status      GET-REPLACE
                                   REPLACE-WITH-DEFAULT
                                   DEFAULT VALUE 0
                                   INITIAL VALUE 0
                                   PERMITTED VALUES
                                   BOOLEAN (0-inactive, 1-active),

          rx-tdm-parity-error-counter
                                   GET-REPLACE
                                   REPLACE-WITH-DEFAULT
                                   DEFAULT VALUE 0
                                   INITIAL VALUE 0
                                   PERMITTED VALUES
                                   INTEGER (0...1023),

          rx-tdm-parity-error-thres
                                   GET-REPLACE
                                   REPLACE-WITH-DEFAULT
                                   DEFAULT VALUE 255
                                   INITIAL VALUE 255
                                   PERMITTED VALUES
                                   INTEGER (255...1023),

          synch-loss-counter       GET-REPLACE
                                   REPLACE-WITH-DEFAULT
                                   DEFAULT VALUE 0
                                   INITIAL VALUE 0,
```

```
        synch-loss-thres        GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 127
                                INITIAL VALUE 127
                                PERMITTED VALUES
                                INTEGER (127...1023),

        synch-status            GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                BOOLEAN (0-no sync, 1-sync),

        parity-check-fail-counter
                                GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0

        parity-check-fail-thres GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 15
                                INITIAL VALUE 15
                                PERMITTED VALUES
                                INTEGER (0...255),

        rem-parity-check-status GET-REPLACE
                                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                BOOLEAN (0-FALSE, 1-TRUE),

    ACTIONS
        initiate-remote-loopback;
        force-parity-check,
        force-PHY-shutdown;

    NOTIFICATIONS
        rem-loopback-failure;
        rx-tdm-parity-error-thres-exceeded;
        synch-loss-thres-exceeded;
        parity-check-fail-thres-exceeded;
        rem-parity-check-status;

    REGISTERED AS { islan phy(1) package(4) ps-char(1) };
```

10.13.2.3 PMD characteristics package

```
pmd-characteristics   PACKAGE

    BEHAVIOR  pmd-char-behavior;
```

```
ATTRIBUTES
       phy-entity-id                  GET,
       manufacturer-id                GET,
       manufacturer-product-id        GET,
       manufacturer-product-version   GET,

       loc-loopback-status            GET-REPLACE
                                      REPLACE-WITH-DEFAULT
                                      INITIAL VALUE 0
                                      DEFAULT VALUE 0
                                      PERMITTED VALUES
                                      BOOLEAN (0-inactive,1-active),

       invalid-decode-counter         GET-REPLACE
                                      REPLACE-WITH-DEFAULT
                                      DEFAULT VALUE 0
                                      INITIAL VALUE 0
                                      PERMITTED VALUES
                                      INTEGER (0...1023),

       invalid-decode-thres           GET-REPLACE
                                      REPLACE-WITH-DEFAULT
                                      DEFAULT VALUE 511
                                      INITIAL VALUE 511
                                      PERMITTED VALUES
                                      INTEGER (0...1023),

       aggregate-trans-speed          GET-REPLACE
                                      REPLACE-WITH-DEFAULT
                                      DEFAULT VALUE 1
                                      INITIAL VALUE 1
                                      RANGE (1...15)
                                      PERMITTED VALUES
                                      INTEGER (1, 5)

ACTIONS
       initiate-local-loopback;

NOTIFICATIONS
       decode-invalid-thr-exceeded;

REGISTERED AS { islan phy(1) package(4) pmd-char(2) };
```

10.14 Management of the ISDN user/network interface in the ISLAN

The ISTE is, from the standpoint of ISDN bearer service and teleservice provision, an ISDN TE. As such it can be subject to ISDN management functions, as defined in CCITT Recommendation Q.940 (1988) and elsewhere. The ISDN management functional actions may originate from within the ISDN telecommunication network or from the user side of the ISDN user/network interface. In any case, there is a major requirement to coordinate OSI and ISDN management architectures and applications, in so far as they are intended to manage the same or different features of one ISTE.

The specification of the reconciliation and harmonization of OSI and ISDN management functions is a matter for further study, as is the definition of the relationship between OSI management, ISDN management, ISDN signalling, and private network signalling within the ISLAN. It is important to note that the IEEE 802.9 ISLAN standard supports provision of ISDN services to the desktop. Thus, there is a requirement of specifying at least some of the MOs' specification which is useful for the ISDN management. It is not within the scope of this standard to specify all of the necessary MOs for ISDN services. The following text provides a general guideline for MO specification for ISDN-related service provision. This text is necessary to understand ISDN service provisioning via IEEE 802.9, but is not sufficient for the overall implementation of ISDN management.

10.14.1 MOs specific to LAPD protocol

The LAPD MOs are contained within an LAPD entity MO. For an understanding of the specification of the MO class and registration methods, refer to the appropriate ISO, CCITT, and ANSI documents on ISDN management. The PHY Layer MOs specified for HMUX, PS, and PMD sublayers in 10.13 are also applicable to ISDN management. In the following subclauses, only LAPD MOs are specified.

10.14.1.1 Generic LAPD characteristics package

The generic LAPD characteristics are inherited by the ISLAN LAPD entity MO.

```
generic-lapd-entity  MANAGED OBJECT CLASS

      DERIVED FROM  GENERIC LAPD:communications entity;
      CHARACTERIZED BY  generic-lapd-characteristics;

      REGISTERED AS { islan lapd(2) managedObjectClass(4)
             generic-lapd(0) };

hmux-characteristics  PACKAGE

      BEHAVIOR   generic-lapd-behavior;

      ATTRIBUTES
             lapd-entity-id                GET,
             manufacturer-id               GET,
             manufacturer-product-id       GET,
             manufacturer-product-version  GET,
             fcs-error-count               GET,
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             invalid-frame-count           GET,
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             max-lapd-size                 GET,
                                           REPLACE WITH DEFAULT
```

```
                                DEFAULT VALUE 256
                                INITIAL VALUE 256
                                REQUIRED VALUE 256,

max-lapd-size-received          GET,
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                  {in accordance with CCITT
                                   Q.921},

lapd-octets-received            GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

lapd-octets-sent                GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

lapd-receive                    GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

lapd-sent                       GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...4294967295),

aged-lapd-discard-count         GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...511),

aged-lapd-discard-thres         GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 127
                                INITIAL VALUE 127
                                PERMITTED VALUES
                                INTEGER (0...511),
```

```
        max-send-lapd-q-age          GET
                                     REPLACE WITH DEFAULT
                                     DEFAULT VALUE 20 ms
                                     INITIAL VALUE 20 ms
                                     PERMITTED VALUES
                                     INTEGER ( 0...127ms),


    ACTIONS
        activate-lapd;
        deactivate-lapd;
        reset-lapd;

    NOTIFICATIONS
        aged-lapd-discard-thres-not;

    REGISTERED AS { islan lapd(2) package(4) generic-lapd(0) };
```

10.14.1.2 ISLAN LAPD address package

The lapd-address characteristics are inherited by the ISDN LAPD entity MO.

```
lapd-address  MANAGED OBJECT CLASS

    DERIVED FROM  GENERIC LAPD:communications entity;
    CHARACTERIZED BY  lapd-address;

    REGISTERED AS { islan lapd(2) managedObjectClass(4)
        lapd-address(1) };

islan-lapd-address-characteristics  PACKAGE

    BEHAVIOR DEFINITIONS  lapd-address-behavior

    ATTRIBUTES
        usable-sapi-control     GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...1023),

        TEI-assign-fail-count   GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...1023),
        TEI-assign-fail-thres   GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 255
                                INITIAL VALUE 255
                                PERMITTED VALUES
                                INTEGER (0...1023),
```

```
TEI-DISC-request-count  GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 0
                        INITIAL VALUE 0
                        PERMITTED VALUES
                        INTEGER (0...1023),

TEI-DISC-request-thres  GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 255
                        INITIAL VALUE 255
                        PERMITTED VALUES
                        INTEGER (0...1023),

link-reestablishment-counter
                        GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 0
                        INITIAL VALUE 0
                        PERMITTED VALUES
                        INTEGER (0...1023),

link-reestablishment-thres
                        GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 31
                        INITIAL VALUE 31
                        PERMITTED VALUES
                        INTEGER (0...1023),

buffer-overflow-counter GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 0
                        INITIAL VALUE 0
                        PERMITTED VALUES
                        INTEGER (0...1023),

buffer-overflow-thres   GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 31
                        INITIAL VALUE 31
                        PERMITTED VALUES
                        INTEGER (0...1023),

lapd-retransmit-counter GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 0
                        INITIAL VALUE 0
                        PERMITTED VALUES
                        INTEGER (0...4294967295),

lapd-retransmit-thres   GET
                        REPLACE WITH DEFAULT
                        DEFAULT VALUE 31
```

```
                                  INITIAL VALUE 31
                                  PERMITTED VALUES
                                  INTEGER (0...4294967295),

             unknown-tei-count    GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...255),

             unknown-tei-thres    GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 31
                                  INITIAL VALUE 31
                                  PERMITTED VALUES
                                  INTEGER (0...255),

             unknown-sapi-count   GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...255),

             unknown-sapi-thres   GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 31
                                  INITIAL VALUE 31
                                  PERMITTED VALUES
                                  INTEGER (0...255),

      NOTIFICATIONS
             tei-assign-fail-thres-not;
             tei-disc-req-thres-not;
             link-reestablishment-thres-not;
             buffer-overflow-thres-not;
             lapd-retransmit-thres-not;
             unknown-tei-receive-thres-not;
             unknown-sapi-receive-thres-not;

      REGISTERED AS { islan lapd(2) package(4) lapd-address(1) };
```

10.14.2 MOs specific to CCITT Q.93x

This subclause provides a set of MOs pertaining to the ISDN CCITT Q.93x protocol layer as applicable to IEEE 802.9 ISLAN architecture. Specific requirements of management functionality at this layer require consideration of all of the aspects of CCITT, ANSI, and ISO for CCITT Q.93x layer management, including specification of MOs to various timers.

The q93x-entity characteristics are inherited by the ISLAN CCITT Q.93x entity MO.

```
q931-entity  MANAGED OBJECT CLASS
      DERIVED FROM GENERIC Q931:communications entity;
```

```
      CHARACTERIZED BY  q931-characteristics;

      REGISTERED AS { islan q931(3) managedObjectClass(4)
             generic-q931(0) };

islan-q93x-operations-characteristics  PACKAGE

      BEHAVIOR DEFINITIONS  q93x-operations-address-behavior

      ATTRIBUTES
             signalling-entity-id          GET,
             manufacturer-id               GET,
             manufacturer-product-id       GET,
             manufacturer-product-version  GET,

             valid-calls-count             GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             invalid-13-msgs-count         GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             invalid-fail-thres            GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 31
                                           INITIAL VALUE 31
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             all-channels-busy-count       GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
                                           INTEGER (0...255),
             all-channels-busy-thres       GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 31
                                           INITIAL VALUE 31
                                           PERMITTED VALUES
                                           INTEGER (0...255),

             invalid-info-elements-count   GET
                                           REPLACE WITH DEFAULT
                                           DEFAULT VALUE 0
                                           INITIAL VALUE 0
                                           PERMITTED VALUES
```

```
                                      INTEGER (0...255),
        invalid-info-elements-thres   GET
                                      REPLACE WITH DEFAULT
                                      DEFAULT VALUE 127
                                      INITIAL VALUE 127
                                      PERMITTED VALUES
                                      INTEGER (0...255),
    NOTIFICATIONS
        invalid-fail-thres-not;
        all-channels-busy-thres-not;
        invalid-info-elements-thres-not;
```

10.14.2.1 MOs specific to call setup phase

```
islan-q93x-callsetup-characteristics  PACKAGE

    BEHAVIOR DEFINITIONS q93x-operations-address-behavior

    ATTRIBUTES
        l3-setup-received-count       GET
                                      REPLACE WITH DEFAULT
                                      DEFAULT VALUE 0
                                      INITIAL VALUE 0
                                      PERMITTED VALUES
                                      INTEGER (0...127),

        l3-setup-aged-count           GET
                                      REPLACE WITH DEFAULT
                                      DEFAULT VALUE 0
                                      INITIAL VALUE 0
                                      PERMITTED VALUES
                                      INTEGER (0...1023ms),

        l3-setup-aged-thres           GET
                                      REPLACE WITH DEFAULT
                                      DEFAULT VALUE 511 ms
                                      INITIAL VALUE 511 ms
                                      PERMITTED VALUES
                                      INTEGER (0...1023ms),
```

MOs for timers must be specified in accordance with the CCITT Recommendations Q.931 (1993) and Q.932 (1993); as such, the details are not within the scope of this standard.

```
    NOTIFICATIONS:

        l3-setup-aged-thres-not;
```

10.14.2.2 MOs specific to call termination phase

```
islan-q93x-calltermination-characteristics      PACKAGE

    BEHAVIOR DEFINITIONS   islan-q93x-calltermination

    ATTRIBUTES
```

```
    l3-disconnect-aged-count      GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...255),

    l3-disconnect-aged-thres      GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 127
                                  INITIAL VALUE 127
                                  PERMITTED VALUES
                                  INTEGER (0...255),

    l3-release-received-counter   GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...127),

    l3-codeset-type               GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  RANGE(0...15)
                                  PERMITTED VALUES
                                  INTEGER (0...7),

    l3-codeset-type-receive-error-count
                                  GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...127),

    l3-codeset-type-receive-error-thres
                                  GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 31
                                  INITIAL VALUE 31
                                  PERMITTED VALUES
                                  INTEGER (0...127),

    l3-information-element-error  GET
                                  REPLACE WITH DEFAULT
                                  DEFAULT VALUE 0
                                  INITIAL VALUE 0
                                  PERMITTED VALUES
                                  INTEGER (0...127),

NOTIFICATIONS
```

```
        l3-disconnect-aged-thres-not;
        l3-codeset-type-receive-error-thres-not;
        l3-information-element-error-not;

    REGISTERED AS { islan q931(3) package(4)* q931-entity(0) };
```

Note: All Q931 attributes in 10.14.2, 10.14.2.1, and 10.14.2.2 belong to this package.

10.14.2.3 MOs specific to information phase

The information channel characteristics are inherited by the ISLAN channel entity MO.

```
isdn-chan-entity  MANAGED OBJECT CLASS

    DERIVED FROM  GENERIC ISDN:communications entity;
    CHARACTERIZED BY  isdn-chan-entity-characteristics;

    REGISTERED AS { islan isdn chan(4) managedObjectClass(4)
        isdn-chan-entity(0) };

islan-information-characteristics  PACKAGE
    BEHAVIOR DEFINITIONS   islan-information

    ATTRIBUTES

        B1-chan-service-type    GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                BOOLEAN (0-circuit mode,
                                 1-packet mode),

        B2-chan-service-type    GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                BOOLEAN (0-circuit mode,
                                 1-packet mode),

        C-channels-allocated-count
                                GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 0
                                INITIAL VALUE 0
                                PERMITTED VALUES
                                INTEGER (0...255),

        C-channels-allocated-thres
                                GET
                                REPLACE WITH DEFAULT
                                DEFAULT VALUE 127
                                INITIAL VALUE 127
```

```
                          PERMITTED VALUES
                          INTEGER (0...255),

      max-bw-Ci-channel   GET
                          REPLACE WITH DEFAULT
                          DEFAULT VALUE 255
                          INITIAL VALUE 255
                          PERMITTED VALUES
                          INTEGER (0...320)
                          {bandwidth constructed in increments
                           of 64 kb/s},

      max-bw-Ci-channel-assigned
                          GET
                          REPLACE WITH DEFAULT
                          DEFAULT VALUE 0
                          INITIAL VALUE 0
                          PERMITTED VALUES
                          INTEGER (0...255)
                          {bandwidth constructed in increments
                           of 64 kb/s},

      min-bw-Ci-channel-assigned
                          GET
                          REPLACE WITH DEFAULT
                          DEFAULT VALUE 0
                          INITIAL VALUE 0
                          PERMITTED VALUES
                          INTEGER (0...255),

      C-channel-service-type GET
                          REPLACE WITH DEFAULT
                          DEFAULT VALUE 0
                          INITIAL VALUE 0
                          PERMITTED VALUES
                          INTEGER (0...15)
                          {0 = circuit mode;
                           1 = packet mode data},

      P-channel-bw        GET
                          REPLACE WITH DEFAULT
                          DEFAULT VALUE 1
                          INITIAL VALUE 1
                          PERMITTED VALUES
                          INTEGER (1...20)
                          { 1 = 1 Mb/s; 2 = 2 Mb/s, ...},
NOTIFICATIONS
      c-channels-allocated-thres-not;

REGISTERED AS { islan isdn-chan(4) package(4) isdn-chan-
      entity(0) };
```

10.15 MIB tables

The following tables provide a summary of the MOs for the IEEE 802.9 ISLAN interface specification (see tables 10-1 through 10-13). The MOs have been grouped into categories for easier inspection (e.g., ISTE vs. AU MOs).

The MO specification requires identification of the current values, default values, and initial values according to the following definition:

Current values	Describes the current state of the system variable.
Default values	Describes the default value that system management will use as the operational value if none other is specified.
Initial values	Describes the operational value that system administration will configure at system startup.
LAN or ISDN Mgr, or both	Describes whether the element is managed by the central LAN management system or the central ISDN management system, or both.
Mandatory or optional	Describes whether the support of this element is mandatory (M) or optional (O).

All counters for the MOs are 32-bit counters. For implementations using one AC field per TDM frame period, all timers shall have 125 μs granularity. For implementations using multiple AC fields, all timers shall have granularity equivalent to the rate of occurrence of AC fields in a TDM frame.

Table 10-1—System-level MOs

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
iste-type-id	–	–	–	Both	M	Specifies the ISTE manufacturer ID, ISTE type and version.
hmux-type-id	–	–	–	Both	M	Specifies the HMUX manufacturer ID, HMUX type and version.
mac-entity-id	–	–	–	LAN	M	Specifies the MAC manufacturer ID, MAC type and version.
phy-entity-id	–	–	–	Both	M	Specifies the PHY manufacturer ID, PHY type and version.
lapd-entity-id	–	–	–	ISDN	M	Specifies the version of ISDN Layer 2.
signalling-entity-id	–	–	–	ISDN	M	Specifies the version of ISDN Layer 3.

Table 10-2—MAC protocol MOs—Specific to MPDU

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
max-mpdu-size	5119	5119	0	LAN	M	Max size in octets.
max-mpdu-size-received	0	0	0	LAN	M	Max MPDU size received in a session.
max-priority-requested	0	0	0	LAN	O	Permitted values: 0 (min) – 7 (max)
min-mpdu-size	19	19	19	LAN	M	Min size in octets.
min-priority-requested	0	0	0	LAN	O	Permitted values: 0 (min) – 7 (max)
mpdu-octets-received-count	0	0	0	LAN	M	Counts the number of octets received continuously. 32-bit counter.
mpdu-octets-sent-count	0	0	0	LAN	M	Counts the number of octets transmitted continuously. 32-bit counter.
mpdu-received-count	0	0	0	LAN	M	Counts the number of MPDUs received over a period of time. 32-bit counter.
mpdu-received-in-error-count	0	0	0	LAN	M	Counts the number of MPDUs received in error. 32-bit counter.
mpdu-received-in-error-thres	0	255	255	LAN	M	Threshold set for management action.
mpdu-sent-count	–	0	0	LAN	M	Indicates how many MPDUs were sent.
priority-last-requested	–	0	0	LAN	O	Priority levels: 0 (min) – 7(max)

Table 10-3—MAC protocol MOS—Specific to resource allocation

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
rx-max-err-count	–	0	0	LAN	M	Specifies error if received MPDU length exceeds 5119 octets. 32-bit counter.
rx-max-err-count-thres	–	255	255	LAN	M	Threshold for management action.
rx-min-err-count	–	0	0	LAN	M	Incremented if length of the received MPDUs is less than 19 octets.
rx-min-err-count-thres	–	255	255	LAN	M	Threshold for management action.
rx-size-last-mpdu	–	0	0	LAN	O	Size of last MPDU received (19–5119).
rx-grant-timer-err-count	–	0	0	LAN	O	Incremented each time Grant-timer has expired.
rx-grant-timer-err-thres	–	16	16	LAN	O	Threshold for management action.
rx-res-err	–	0	0	LAN	O	An error flag that specifies that SOF has been received but no receive buffers are available.
max-req-time	–	0	0	LAN	M	Indicates how long a request is asserted without having received a grant. 32-bit counter.
max-req-err-count	–	0	0	LAN	M	Incremented each time the tx-req-timer has expired. 32-bit counter.
max-req-err-count-thres	–	0	0	LAN	M	Threshold for management action. 32-bit counter.

Table 10-4—MAC protocol MOs—Core ISLAN MAC characteristics

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
max-grant-time	–	100 ms	100 ms	LAN	M	Default = 10x (max frame transmission time for 4 Mb/s) and 20x (max frame transmission time for 20 Mb/s).
mpdu-to-send	–	0	0	LAN	M	0 = False, 1 = True MPDU is ready to be sent (value = 1).
hmux-mac-address-enable	–	0	0	LAN	M	This is for MAC filtering. 0 = Enable, 1 = Disable

Table 10-5—MAC protocol MOs—Specific to ISTE MAC

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
iste-aged-mpdu-discard-count	–	0	0	LAN	O	Permitted value: 0–127 MPDU may be discarded or treated as new.
iste-aged-mpdu-discard-thres	–	127	127	LAN	O	Threshold for management action.
iste-max-send-mpdu-q-age	–	20 ms	20 ms	LAN	O	Value: 0–27 ms
iste-ignore-grant	–	0	0	LAN	M	0 = Not ignore grant 1 = Ignore grant

Table 10-6—MAC protocol MOs—Specific to AU MAC

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
au-aged-mpdu-discard-count	–	0	0	LAN	O	Permitted value: 0–511 MPDU may be discarded or treated as new.
au-aged-mpdu-discard-thres	–	511	511	LAN	O	Threshold for management action.
au-max-send-mpdu-q-age	–	20 ms	20 ms	LAN	O	Value: 0–127 ms
au-ignore-grant	–	1	1	LAN	M	0 = Not ignore grant 1 = Ignore grant AU ignores grant for request/grant protocol.

NOTE—The MOs for the AU's MIB are a matter for further study.

Table 10-7—MOs specific to PHY Layer—HMUX sublayer

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
tx-mode-control	–	0	0	Both	M	Range of values: 0– 7; Modes 0, 1, 2, 3 are specified.
rx-mode-status	–	0	0	Both	M	Range of values: 0–7; Modes 0, 1, 2, 3 are specified.
mode	–	0	0	Both	M	Mode 0 is the default. Modes 0, 1, 2, 3 are specified.
mf-error-count	–	0	0	Both	M	Number of MF errors. Mandatory use only in Mode 3.
mf-error-thres	–	16	16	Both	M	MF error threshold. Mandatory use only in Mode 3.
d-chan-rate-control	–	16 kb/s	16 kb/s	Both	M	Specifies rate supported (16 kb/s or 64 kb/s). Used with DCR field.
d_chan-rate-status	–	16 kb/s	16 kb/s	Both	M	Specifies rate supported (16 kb/s or 64 kb/s).
nb-AC-fields	–	1	1	Both	M	Specifies number of AC fields in the TDM frame. Permitted values: 1–5.
secure-control	–	0	0	Both	O	Specifies secure field. 0 = No security 1 = Invoke security

Table 10-8—MOs specific to PHY Layer—PS sublayer

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
rem-loopback-status	–	0	0	Both	M	Specifies status of remote loopback. 0 = Inactive, 1 = Active
rx-tdm-parity-error-counter	–	0	0	Both	M	Range: 0–1023.
rx-tdm-parity-error-thres	–	255	255	Both	M	Permitted values: 0–1023.
synch-loss-counter	–	0	0	Both	M	Specifies number of sync failures.
synch-loss-thres	–	127	127	Both	M	Range: 0–1023.
synch-status	–	0	0	Both	M	0 = No sync, 1 = Sync
parity-check-fail-counter	–	0	0	Both	O	Specifies number of parity check failures.
parity-check-fail-thres	–	15	15	Both	O	Permitted values: 0–255.
rem-parity-check-status	–	0	0	Both	M	0 = False 1 = True

Table 10-9—MOs specific to PHY Layer—PMD sublayer

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
loc-loopback-status	–	0	0	Both	M	Specifies status of local loopback. 0 = Inactive, 1 = Active
invalid-decode-counter	–	0	0	Both	M	Keeps track of missed PMD synchronization. Range: 0–1023.
invalid-decode-thres	–	511	511	Both	M	Threshold for missed synchronization counter.
aggregate-trans-speed	–	1	1	Both	M	Range: 1–15 with each integer representing N × (4Mb/s) bandwidth. Permitted values: N = 1,2,3,4,5. Higher values are a matter for further study.

Table 10-10—MOs specific to LAPD—Generic LAPD

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
fcs-error-count	–	0	0	ISDN	M	FCS bad frame count threshold. Range: 0–255.
invalid-frame-count	–	0	0	ISDN	M	Invalid frame count threshold. Range: 0–255.
max-lapd-size	–	256	256	ISDN	M	In accordance with CCITT Recommendation Q.921. Indicates max size of LAPD
max-lapd-size-received	–	0	0	ISDN	O	Indicates max LAPD size received.
lapd-octets-received	–	0	0	ISDN	M	Counts the number of octets received. 32-bit counter.
lapd-octets-sent	–	0	0	ISDN	M	Counts the number of octets sent. 32-bit counter.
lapd-received	–	0	0	ISDN	M	Counts the number of LAPDs received.
lapd-sent	–	0	0	ISDN	M	Indicates how many LAPDs have been sent. 32-bit counter.
aged-lapd-discard-count	–	0	0	ISDN	O	Permitted values: 0–511. Aged LAPD frame may be discarded or treated as new.
aged-lapd-discard-thres	–	127	127	ISDN	O	Threshold for management action. Range: 0–511.
max-send-lapd-q-age	–	20 ms	20 ms	ISDN	O	Value: 0–127 ms.
all timers-Q.921	–	–	–	ISDN	M	All MOs used with timers shall be in accordance with CCITT Recommendation Q.921.

Table 10-11—MOs specific to LAPD—LAPD address

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
usable-sapi-control	–	0	0	ISDN	M	Definition of SAPI values. Range: 0–1023.
tei-assign-fail-count	–	0	0	ISDN	M	Counts the number of times the assignment of TEI is not possible. Range: 0–1023.
tei-assign-fail-thres	–	255	255	ISDN	M	Threshold for management action. Range: 0–1023.
tei-disc-request-count	–	0	0	ISDN	M	Counts the number of requests for link termination. Useful for link failure identification.
tei-disc-request-thres	–	255	255	ISDN	M	Threshold for the above counter.
link-reestablish-ment-counter	–	0	0	ISDN	O	Counts the number of times the Data Link Layer is reestablished.
link-reestablish-ment-thres	–	31	31	ISDN	O	Threshold for management action. Range: 0–1023.
buffer-overflow-counter	–	0	0	ISDN	O	Counts the number of receive frame overflows. Range: 0–1023.
buffer-overflow-thres	–	31	31	ISDN	O	Threshold for management action.
lapd-retransmit-counter	–	0	0	ISDN	O	Counts the LAPD frame retransmissions. 32-bit counter.
lapd-retransmit-thres	–	31	31	ISDN	O	Threshold for management action.
unknown-tei-count	–	0	0	ISDN	O	Useful for monitoring the number of wrong TEIs being received.
unknown-tei-thres	–	31	31	ISDN	O	Threshold for management action. Range: 0– 255.
unknown-sapi-count	–	0	0	ISDN	O	Useful for monitoring the number of unknown LAPD frames.
unknown-sapi-thres	–	31	31	ISDN	O	Threshold for management action.

Table 10-12—MOs specific to ISDN CCITT Q.931 protocol

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
valid-calls-count	–	0	0	ISDN	O	Useful for resource mgmt.
invalid-l3-msgs-count	–	0	0	ISDN	M	Counts the no. of invalid SET-UP messages from ISTE to NT. See Note 1.
invalid-fail-thres	–	31	31	ISDN	M	Threshold for mgmt. action.
all-channels-busy-count	–	0	0	ISDN	O	Counts the no. of attempts to set up a call, but all channels in ISDN network are busy.
all-channels-busy-thres	–	127	127	ISDN	O	Threshold for mgmt. action.
invalid-info-elements-count	–	0	0	ISDN	M	Counts the no. of invalid info. elements for alarm.
invalid-info-thres	–	127	127	ISDN	M	Threshold for mgmt. action.
l3-setup-received-count	–	0	0	ISDN	M	Counts the no. of setup messages received.
l3-setup-aged-count	–	0	0	ISDN	M	No setup acknowledgments received (granularity in ms).
l3-setup-aged-thres	–	511	511	ISDN	M	Threshold for mgmt. action (granularity in ms).
Q.931 timers	–			ISDN	O	All MOs associated with timers for call control shall be used in accordance with the CCITT Q.93x Recommendations. See Note 2.
l3-disconnect-aged-count	–	0	0	ISDN	O	No release has been received.
l3-disconnect-aged-thres	–	127	127	ISDN	O	Threshold for mgmt. action.
l3-release-received-counter	–	0	0	ISDN	O	1 = True (release received), 0 = False (release not received)
Q.93x timers	–				M	All MOs associated with timers for call control shall be used in accordance with the CCITT Q.93x Recommendations.
l3-codeset-type	–	0	0	ISDN	M	Identifies the current codeset in use. Range 0–15. Codeset 0 and Codeset 7 are permitted.
l3-codeset-type-receive-error-count	–	0	0	ISDN	M	Codeset received in error or unknown codeset.
l3-codeset-type-receive-error-thres	–	31	31	ISDN	M	Threshold for mgmt. action.
l3-information-element-error	–	0	0	ISDN	M	Identifies information element in error.

NOTES

1—A double counting takes place between invalid CCITT Q.93x messages and invalid information elements. This is because interworking problems are anticipated with respect to erroneous codeset usage.

2—CCITT Q.93x refers to the use of the protocol family which includes CCITT Q.931, Q.932, and Q.933.

Table 10-13—MOs specific to ISDN channels

Managed object (MO)	Current value	Default value	Initial value	LAN or ISDN Mgr, or both	Mandatory or optional	Comments
B1-chan-service-type	–	0	0	ISDN	O	Usage of B1 channel. 0 = Circuit mode 1 = Packet mode
B2-chan-service-type	–	0	0	ISDN	O	Usage of B2 channel. 0 = Circuit mode 1 = Packet mode
C-channels-allocated-count	–	0	0	ISDN	M	Specifies number of C channels allocated. Range: 0–320 where 0=none.
C-channels-allocated-thres	–	127	127	ISDN	M	Specifies the max number of C channels that can be assigned. Range: 0–320.
max-bw-Ci-channel	–	255	255	ISDN	M	Max bandwidth permitted for any given C channel. Range = 0–320.
max-bw-Ci-channel-assigned	–	0	0	ISDN	O	Max bandwidth of a currently assigned C channel. Where n = 0–320.
min-bw-Ci-channel-assigned	–	0	0	ISDN	O	Min bandwidth of a currently assigned C channel. Where n = 0–320.
C-channel-service-type	–	0	0	ISDN	O	Identifies circuit mode (0).
P-channel-bw	–	1	1	Both	O	Identifies the current P channel bandwidth. Range 1–15, where it is $n \times 64$ kb/s.

11. Signalling and ISDN management scope

11.1 Introduction

This chapter, in conjunction with annex D, conveys the recommended signalling requirements for the IEEE 802.9 architecture. Signalling specifies the procedures for establishing, maintaining, and clearing calls between the IS user application (or TE) and the network or access unit (AU). It conveys information that identifies which potential payload time slots (octets) in the IEEE 802.9's TDM frame are to be allocated to specific B, C, or P channel service access points (SAPs). Therefore, a recommended set of signalling procedures is required for the coordination of connection management between the ISTE and the IEEE 802.9 AU.

The signalling procedures are defined in terms of messages and information elements carried in these messages, which are exchanged over the D channel, for creation, monitoring, and deletion of point-to-point variable bandwidth channels. These messages provide the stimuli for provisioning dynamically C and B channels, by signalling the station management entities within the ISTE and the AU. These management entities, in turn, signal the hybrid multiplexer (HMUX) template setup, enabling the embedded channels and coordinating the two network management procedures. The station management entities in the ISTE and the AU are in fact management agents capable of dialog with an IEEE 802 style LAN management. The station management entity signals the control center AU for the purpose of configuration, accounting, fault, performance, and security management functions as shown in figure 11-1.

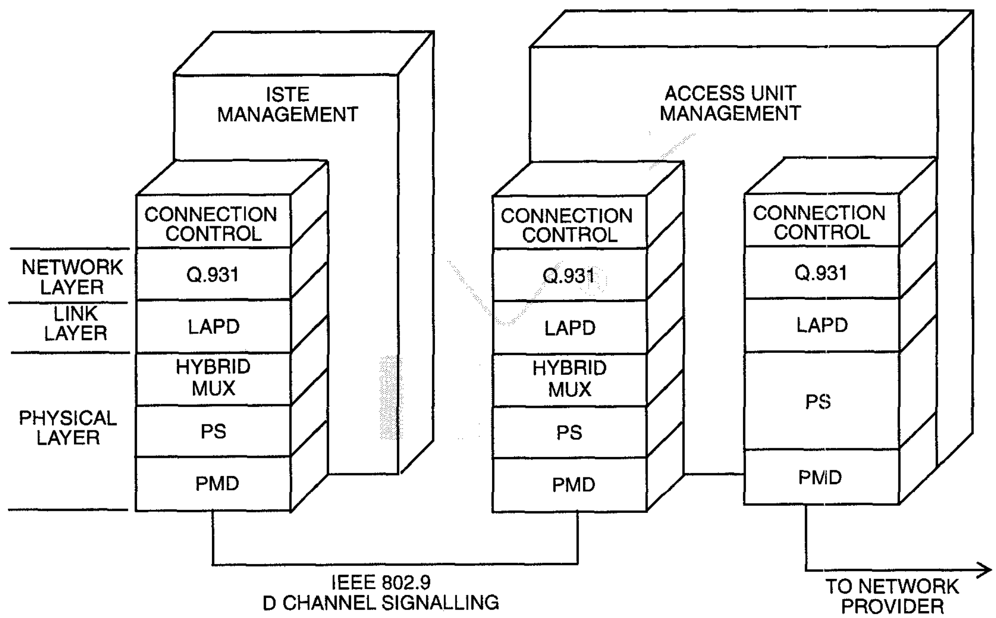


Figure 11-1—Signalling protocol model

In providing signalling for the IEEE 802.9 architecture, this standard uses CCITT Recommendation Q.931 (1993) as its basic message structure. CCITT Recommendations Q.930 (1993) and Q.931 (1993) have been developed and accepted by the International Telecommunications Union (ITU-T) as a Layer 3 call control protocol for configuration management of ISDN networks. This chapter and annex D recommend the utilization of CCITT Q.931 and a methodology of information element extensions to CCITT Q.931 Codeset 0. Those extensions and message structures are the subject of this chapter and annex D.

As shown in figure 11-1, CCITT Q.931 utilizes the services provided by the Data Link Control (DLC) Layer, which have been defined in CCITT Recommendations Q.920 (1993) and Q.921 (1993). Link access procedure on the D channel (LAPD) provides functions for establishment of data link connections, error protected in-sequence transmission of data, and reestablishment of data link connections.

The CCITT Q.931 ISTE management entity is responsible as a management agent for:

- Negotiation of bandwidth (configuration) management, fault management, performance management, and security management of the local interface between the ISTE and the AU
- Negotiation of service provisioning over the local ISTE interface in order to access both LAN and ISDN.

The ISLAN AU will need to negotiate with the ISDN network for the provisioning of service in order to access the wide area network (WAN). The types of service the AU may negotiate on behalf of the ISTE are as follows:

- Isochronous circuit mode for the transport of voice/data/video for the B and C channels as they are transported across the ISDN WAN
- Isochronous packet mode for the transport of data for the B, C, and D channels as they are transported across the ISDN WAN
- ISDN management information in support of the negotiation and status of services provided for the B, C, and D channels

Therefore, the ISTE must support the CCITT Q.931 management protocol for establishment and maintenance of transmission services. The AU must support the local subnetwork isochronous management functions. The AU must also provide a gateway function allowing adaption (transformation) from the local IEEE 802.9 subnetwork into the ISDN WAN. The definition of the extent of this ISDN interworking function is a matter for future study.

In utilizing the CCITT Q.931 information element extensions in such a manner, IEEE 802.9 signalling remains compatible with existing international network basic signalling methods, as specified by ANSI T1S1 and CCITT Study Group XI, thereby allowing the backbone ISDN network to provide some of the switching responsibility for limited architecture AU products. For more sophisticated AU architectures, the extended message set allows for a superset of signalling methods to be localized within the IEEE 802.9 subnetwork. These superset messages would provide multimedia interconnections between ISTEs, providing dynamic bandwidth call connections to certain WAN providers that provide such extended services.

11.2 General overview of the service

CCITT Q.931 is a "Layer 3" protocol which provides the means to establish, maintain, and terminate network connections across an ISDN network between communication application entities. As such, it provides the user-to-network signalling function particular to certain user applications that require the following functions:

- Process primitives to and from the Data Link Control (DLC) Layer protocol
- Generation and interpretation of Layer 3 peer-to-peer messages
- Administration of timers and all reference values used in call control procedures
- Administration of access resources for both physical and logical channels

As a "Network Layer" protocol, CCITT Q.931 is a simple protocol with respect to its own error recovery and is dependent upon the LAPD protocol, as defined in CCITT Recommendations Q.920 (1993) and Q.921 (1993), to provide an acknowledged information transfer service which includes:

— Provision and identification of data link connections (C and B channels)
— Frame sequence integrity of the data link PDUs
— Notification of unrecoverable errors
— Flow control of the link-to-link endpoints
— Re-establishment of data link connections

The IEEE 802.9 signalling procedures described in this chapter shall use CCITT Recommendations Q.920 (1993) and Q.921 (1993) as an LAPD protocol to provide the above functions between the D channel AC point of the HMUX and the CCITT Q.931 network service layer. These procedures will enable the IEEE 802.9 ISLAN to establish circuit-switched B and C channel connections.

Figure 11-2 shows the typical message transfers which would take place during call setup, information transfer, and call clearing of circuit-switched connections. This chapter identifies a set of CCITT Q.931 messages and their associated information elements to achieve this new isochronous LAN service. A new set of information elements are introduced to support identification of channels, multimedia communications, and the dynamic bandwidth negotiation features of IEEE 802.9. The details of these extensions are provided in this chapter and in annex D.

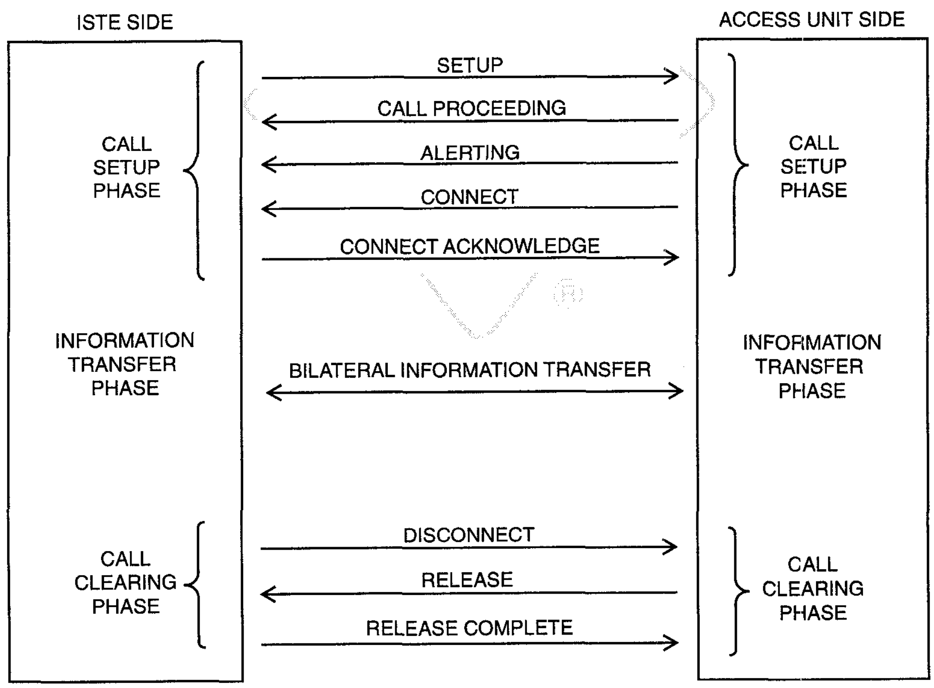


Figure 11-2—Typical CCITT Q.931 message transfers

11.3 Requirements for signalling support

The following subclauses describe the needs for "extensions" of the CCITT Q.931 protocol to apply it for bandwidth management of the various channel types that may be effected in the IEEE 802.9 ISLAN interface.

11.3.1 B channel signalling requirements

The following observation may be made with respect to the application of CCITT Q.931 to establish B channels:

a) Each B channel may be established as a separate bearer channel.
b) Each B channel may be established as a circuit-switched or a packet-switched bearer channel.

Since the IEEE 802.9 B1 and B2 channels are analogous to the two B channels of an ISDN BRI, no special call control extensions are needed by IEEE 802.9. These functions can be accomplished with the standard CCITT Recommendation Q.931 (1993) Codeset 0 signalling messages.

NOTE—The only extensions would be those employed for the case of packet mode transport using the B channel. Refer to CCITT Recommendation Q.933 (1993) for an example.

11.3.2 D channel signalling requirements

The following observations may be made with respect to the application of CCITT Q.931 protocol to establish packet mode transport over the D channel:

a) The D channel may be used as a transport pipe for call control information (SAPI 0 usage).
b) The D channel may be employed as a packet data transport:
    1) SAPI 1: New packet mode service (frame relay).
    2) SAPI 16: CCITT X.25 packet service.

For signalling purposes, the D channel is identified in the same manner as an ISDN BRI D channel. Special call control extensions are not needed by IEEE 802.9. These functions can be accomplished with the standard CCITT Recommendation Q.931 (1993) (Codeset 0) signalling messages.

11.3.3 C channel signalling requirements

The following observations may be made with respect to the application of CCITT Q.931 (1993) to establish C channels:

a) C channels may be established as circuit-switched or packet-switched bearer channels.
b) Extensions to Q.931 are required to support bearer capability needs.
c) Present information rate field to support and dynamically provision rates of 64 kb/s, 128 kb/s, 384 kb/s, 1544 kb/s, and 1920 kb/s.
    1) C channel(s) require the flexibility of $n \times 64$ kb/s provisioning, and bandwidth negotiation
    2) Identifying multiple C channels
    3) Extend the diagnostics in the cause information element to support the enhanced bearer capability and multiple channel identification needs

The signalling procedures for establishing and releasing C channel calls are described in 11.4 and 11.5.

11.3.4 P channel signalling requirements

The following observations may be made with respect to the application of CCITT Recommendation Q.931 (1993) to establish the P channel:

a) Ordinarily, the P channel will be independent of CCITT Q.931 signalling procedures, although the bandwidth provisioned in the P channel will be affected by the C channels established in the ISTE to AU interface.

b) If the network configuration dictates that the packet channel transport requires explicit bandwidth reservation for the P channel, then CCITT Q.931 procedures can be applied to establish and release P channel calls. Furthermore, the P channel bandwidth can be dynamically adjusted by using the CCITT Recommendation Q.932 (1993) HOLD/RETRIEVE procedures.

c) The required extensions to CCITT Q.931 are the same as C channel signalling requirements, with the addition of support for IEEE 802.9 packet mode bearer capability.

11.4 CCITT Q.931 architecture overview

This clause provides an overview of CCITT Q.931 and shows how CCITT Q.931, with minimal extensions, is used for signalling within the IEEE 802.9 ISLAN. For explicit details on this protocol, refer to CCITT Recommendation Q.931 (1993). Within this protocol, every message shall consist of the following parts:

a) Protocol discriminator
b) Call reference
c) Message type
d) Other information elements, as required (see annex D)

Information elements a) through c) are common to all messages and shall always be present, while information element d) is specific to each message type. It is within the set of information elements d) that IEEE 802.9 will recommend modest extensions to support the 4096 kb/s and 20 480 kb/s rates of the ISLAN protocol. The basic structure of the message format is shown below. This chapter follows standard CCITT Q.931 bit-ordering (terminology), where the least significant bit (LSb) to most significant bit (MSb) are numbered 1 through 8.

11.4.1 Protocol discriminator

The purpose of the protocol discriminator is to distinguish messages for user-network call control from other Layer 3 messages. It also distinguishes messages from those of the OSI network layer protocol units. The protocol discriminator is the first part of every CCITT Q.931 message. The coding in figure 11-3 indicates the call control messages of CCITT Q.931 that are applicable to the IEEE 802.9 standard messaging. Refer to CCITT Recommendation Q.931 (1993), Section 4.2 for more details.

11.4.2 Call reference

The purpose of the call reference is to identify the call at the local ISTE-to-AU interface to which the particular messages applies. The call reference does not have end-to-end significance across ISTE-to-ISTE connections. The call reference is the second part of every message. The call reference is coded as shown in figure 11-3. Refer to CCITT Recommendation Q.931 (1993), Section 4.3 for more details.

11.4.3 Message types

The purpose of the message type is to identify the function of the CCITT Q.931 signalling message being sent. The message type is the third part of every CCITT Q.931 signalling message as illustrated in

8							1	Octets
Protocol Discriminator								
0	0	0	0	1	0	0	0	1
0	0	0	0	Length of call reference value (in octets)				2
Call reference value								3
0	Message type							4
Other information elements								5

Figure 11-3—General message structure

figure 11-3. The messages necessary for call establishment and clearing over an ISLAN are shown in figures 11-4a and 11-4b. A detailed description of each message, which shows the information elements used by each message, is provided in annex D. These messages constitute the minimal subset of CCITT Q.931 that is required for IEEE 802.9 operation. The same procedures as described in CCITT Recommendation Q.931 (1993) apply overall for call establishment and termination, except that during the call setup phase, only enbloc sending is used. In en-bloc sending, the SETUP message contains all of the call information necessary for call establishment. Unless specifically mentioned, all other procedures are the same as those defined in CCITT Recommendation Q.931 (1993), Section 5, and are used with the subset of messages and information elements which are described in this document. Annex D also describes the various call states associated with the sequence of CCITT Q.931 message transfers for the establishment and release of C channels.

11.4.4 Information elements

Figure 11-5 lists the information elements that are to be used with the above-mentioned messages. The TDM frame structure of ISLAN consists of either 64 payload time slots for the interface at 4096 kb/s, or 320 timeslots for the interface at 20 480 kb/s. Two new information elements called the extended bearer capability and the extended channel identification have been introduced to accommodate the isochronous bandwidth/information rates supported by the ISLAN. These two information elements are used to replace the CCITT Q.931 bearer capability and channel identification information elements, for call setup or clearing of C channel calls. The bearer capability and channel identification information elements are used for setup and teardown of calls over the B1 and B2 channels provided in the ISLAN TDM frame, and are therefore listed in figure 11-5. Furthermore, a new diagnostics field for the cause information element for cause numbers 57, 58, 65, and 89, to support the extended bandwidth needs of the ISLAN, is also introduced. These new information elements are described in annex D. All of the other information elements identified follow the same format and encoding as described by CCITT Recommendations Q.931 (1993) and Q.932 (1993).

11.4.4.1 Codeset usage

11.4.4.1.1 General definition

CCITT's specification of the digital subscriber signalling system no. 1 (DSS 1), commonly referred to as the CCITT Q.93x series of signalling protocols, makes use of information elements to specify individual components of call establishment/release messages. Within each information element, one octet is used to specify or "identify" the type of element. The encoding rules have defined a maximum of eight single-octet information elements and a maximum of 128 variable-length information elements. Within the single-octet information elements, there is an element, shift, that is used as a mechanism to expand this existing set of coding points to a set of eight codesets, as shown in figure 11-6. One common value (coding point) in the single-octet information element format is employed in each of the eight codesets to facilitate shifting (changing) from one codeset to another. The contents of the shift information element identify the codeset to be used for the next information element(s). The codeset in use at any given time is referred to as the "active codeset." By convention, Codeset 0 is the initially active codeset.

Bits								
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**Message types**
0	0	0	–	–	–	–	–	*Call establishment messages*
			0	0	0	0	1	Alerting
			0	0	0	1	0	Call proceeding
			0	0	1	1	1	Connect
			0	1	1	1	1	Connect acknowledge
			0	0	1	0	1	Setup
0	1	0	–	–	–	–	–	*Call clearing messages*
			0	0	1	0	1	Disconnect
			0	1	1	0	1	Release
			1	1	0	1	0	Release complete
0	1	1	–	–	–	–	–	*Miscellaneous messages*
			1	1	1	0	1	Status*
			1	0	1	1	1	Status enquiry

*The STATUS message is used for transmitting error management/recovery information between the ISTE and the AU.

a) CCITT Q.931 message type

Bits								
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**Message types**
0	0	0	–	–	–	–	–	*Call information phase messages*
			0	0	1	0	0	Hold
			0	1	0	0	0	Hold acknowledge
			1	0	0	0	1	Retrieve
			1	0	0	1	1	Retrieve acknowledge

b) Optional CCITT Q.932 message types employed in explicit P channel reservation

Figure 11-4—CCITT Q.93x messages

Bits								
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**Information elements**
1	0	0	1	–	–	–	–	Shift
0	0	0	0	0	1	0	0	Bearer capability (Note 1)
0	0	0	0	0	1	0	1	Extended bearer capability (Note 2)
0	0	0	0	1	0	0	0	Cause
0	0	0	1	0	1	0	0	Call state
0	0	0	1	1	0	0	0	Channel identification (see Note 1)
0	0	0	1	1	0	0	1	Extended channel identification (see Note 2)
0	1	1	0	1	1	0	0	Calling party number
0	1	1	0	1	1	0	1	Calling party subaddress
0	1	1	1	0	0	0	0	Called party number
0	1	1	1	0	0	0	1	Called party subaddress
0	1	1	1	1	1	0	0	Low layer compatibility
0	1	1	1	1	1	0	1	High layer compatibility
0	1	1	1	1	1	1	0	User-to-user

NOTES
1—This information element is used for setting up B1 and B2 channel calls.
2—This information element is used for setting up C channel calls, and is a new information element.

Figure 11-5—Information element identifier coding

The application intent of this architecture is to allow special information elements to be defined and used among telecommunication equipment in special instances of private/public switching domains (i.e., subnetworks of a given service provider or national telecommunication network providers).

The assignment of the codeset identification field in the shift information element is as follows:

— Codeset 5 is reserved for information elements reserved for national use. As such they do not have significance across an international boundary. Therefore, Codeset 5 information elements shall be processed at the first signalling exchange beyond the international boundary, unless there are bilateral agreements to the contrary.

— Codeset 6 is reserved for information elements specific to the local network (either private or public). As such they do not have significance across the boundaries between local subnetworks, or across either a national or international boundary. Therefore, Codeset 6 information elements shall be handled according to the procedures for unrecognized information elements beyond the local network boundary, unless further standardization by ANSI or ITU-T or both allows the recognition of Codeset 6 information elements.

Bits			
3	**2**	**1**	**Codesets**
0	0	0	Coseset 0—Initially active codeset for basic CCITT Q.931 elements
0	0	1	Codeset 1—Reserved
0	1	0	Codeset 2—Reserved
0	1	1	Codeset 3—Reserved
1	0	0	Codeset 4—Reserved
1	0	1	Codeset 5—Information elements for national use
1	1	0	Codeset 6—Information elements specific to the local network (private/public)
1	1	1	Codeset 7—User-specific information elements

Figure 11-6—Codeset encoding

— Codeset 7 information elements shall be handled according to the procedures for unrecognized information elements by the first signalling exchange in the local network.

Two codeset shifting procedures are supported:

a) *Locking shift.* The locking shift procedure causes a permanent shift to another codeset. This specified codeset shall remain inactive until another locking shift information element is encountered that specifies the use of another codeset. This procedure is used only to shift to a higher codeset than the one that was left.

b) *Nonlocking shift.* The nonlocking shift procedure provides a temporary shift to the specified lower or higher codeset. The nonlocking shift procedure acts as a delimiter to request that the immediately following information element be processed according to the definition of that codeset. After the interpretation (processing) of the next information element, the active codeset (Codeset 0) is again used for interpreting any following information elements. If a nonlocking shift to Codeset 6 is encountered, only the next information element is interpreted according to the information element identifiers assigned in Codeset 6.

11.4.4.1.2 Use of codesets across IEEE 802.9 subnetworks

The IEEE 802.9 interface affords the support of the D channel which enables CCITT Q.931 messages to be transported between ISTE devices and the AU. The IEEE 802.9 interface specifies the use of standard CCITT Q.931 messages for establishment/release procedures utilized in the dynamic bandwidth management of its payload field for the definition of C channels (located in the payload field of the IEEE 802.9 TDM frame). Associated with these standard CCITT Q.931 messages are standard information elements (i.e., called party and calling party) as well as some information elements that have been modified to accommodate the needs of the IEEE 802.9 ISLAN environment. The special information elements proposed are:

a) Extended bearer capability
b) Extended channel identification
c) Cause (with modified diagnostic numbers)

Annex D contains a description of the structure of these new information elements, together with procedures for their use.

It is proposed that Codeset 7 be used across the IEEE 802.9 subnetwork to house these new elements. The reason that Codeset 7 has been selected is that Codeset 6 has already been actively used between ISPBX equipment and its private terminal endpoints. The use of Codeset 7 allows better differentiation of signalling traffic at the AU between:

— That traffic used solely for bandwidth management between the IEEE 802.9 terminal and terminal adaptor equipment

— That traffic transparently extended from ISPBX across the AU and the terminal adaptor (TA) to some existing ISDN equipment that is connected via a TA.

Figure 11-7 illustrates this application topology which has influenced the specification of Codeset 7 as the private codeset choice for IEEE 802.9 to use across its local subnetwork to extend the CCITT Q.931 information element specification.

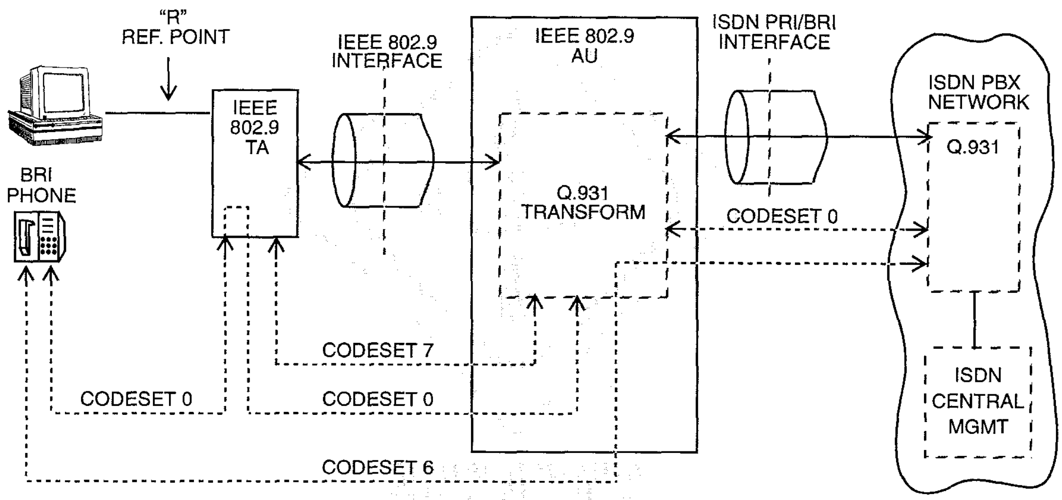


Figure 11-7—Example of codeset usage across IEEE 802.9 subnetwork

With reference to figure 11-7, Codeset 7 is used by the IEEE 802.9 TA to communicate with the IEEE 802.9 AU for the purpose of allocation of bandwidth. The role of the IEEE 802.9 TA is to use Codeset 0 as part of the bandwidth allocation procedures as well. The IEEE 802.9 TA has a responsibility to convey native mode CCITT Q.931 Codeset 0 information which has been originated or is destined for the BRI equipment served by it toward/from the AU. The ISDN facility (indicated as ISDN PRI/BRI in figure 11-7), that connects the ISDN network switch to the AU will utilize Codeset 0 messages for communication of signalling intent to both the AU and the station equipment that may be located behind the AU. If the ISDN switching equipment were in the practice of using Codeset 6 messages for communication with station equipment (native mode ISDN), then the AU could recognize this and simply allow these messages to flow transparently across the AU toward the station equipment. The architecture of the AU and the procedures incorporated within it for purpose of communication with the ISDN network equipment are not within the scope of this standard.

11.4.5 C channel call control procedures

Figures 11-8 and 11-9 show the typical sequence of CCITT Q.931 message transfers for call setup and call clearing of C channel calls. The procedures employed are the same as those specified in CCITT Recommendation Q.931 (1993). The following subclauses describe examples of call setup and call clearing.

11.4.5.1 Call setup

Figure 11-8 shows a call setup message transfer sequence diagram. The ISTE side transmits a SETUP message to the AU using en-bloc sending. In en-bloc sending, the SETUP message contains all of the call information necessary for call establishment, including the extended bearer capability information element. This information element specifies the maximum transfer rate required for the call and optionally may specify a minimum transfer rate for bandwidth negotiation. Optionally, if the user wants to identify the TDM frame time slots to be used, the SETUP message would contain an extended channel identification information element specifying the TDM frame payload time slots required for the maximum information transfer rate requested by the call.

Upon receipt of the SETUP message, the AU checks for availability of the bandwidth and other resources. If the full requested bandwidth is available, the AU reserves this, so that it is not allocated to another user, and sends a SETUP message to the called ISTE for that bandwidth, which includes an extended channel identification information element. A CALL PROCEEDING with an extended channel identification information element is sent to the calling ISTE by the AU. This extended channel identification information element not only defines the TDM frame payload time slots to be used for that call, but also implicitly defines the bandwidth that has been allocated for that call. If there are insufficient bandwidth/resources to meet the request, and the calling ISTE has specified a minimum information transfer rate, the AU will try to allocate bandwidth with a value that is closest to the maximum requested, and is not lower than the minimum. Furthermore, if the calling ISTE had specifically identified the TDM frame payload time slots in the SETUP message, then the AU will allocate a subset of those time slots for the requested call. If the call request cannot be met, the AU sends a RELEASE COMPLETE message back to the calling ISTE.

The called ISTE, upon receipt of the SETUP message, transmits an ALERTING message back to the AU which in turn sends it to the calling ISTE, signifying that called ISTE alerting has been initiated. When the called ISTE is ready to accept the call, it allocates the bandwidth and sends a CONNECT message back to the AU.

The AU, on receipt of a CONNECT message from the called ISTE, allocates the bandwidth and sends a CONNECT ACKNOWLEDGE message back to the called ISTE. The AU then sends a CONNECT message to the calling ISTE, who in turn allocates the bandwidth and responds with a CONNECT ACKNOWLEDGE. End-to-end data transfer can then commence.

11.4.5.2 Call clearing

Figure 11-9 shows the messages exchanged during call clearing. Normal call clearing procedures as described in CCITT Recommendation Q.931 (1993) are employed. The ISTE side transmits a DISCONNECT message to the AU, with the cause information element giving the reason for clearing the call. The AU in turn transmits this message to the destination ISTE, which responds with a RELEASE message. The RELEASE message is conveyed by the AU to the originating ISTE. A RELEASE COMPLETE is then transmitted, acknowledging receipt of the RELEASE message.

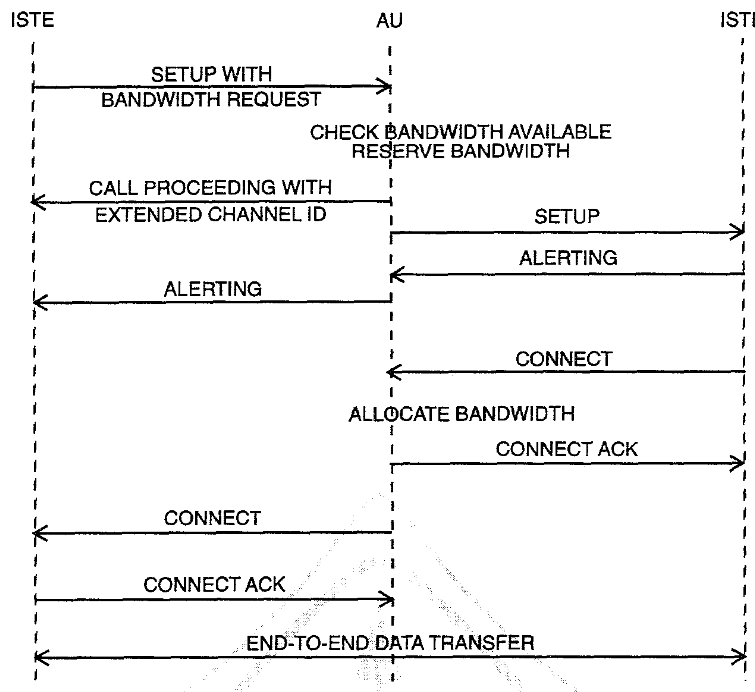


Figure 11-8—Call setup sequence diagram

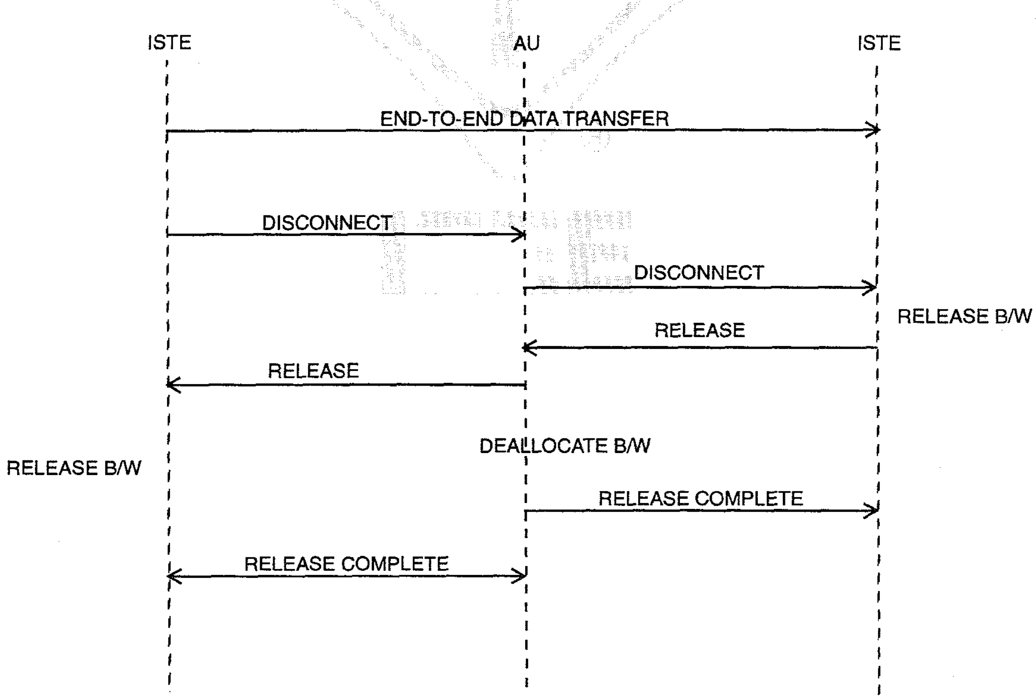


Figure 11-9—Call clearing sequence diagram

11.4.6 User-to-user signalling

Implicit service 1 user-to-user signalling is supported, which provides a way for users to communicate by transferring user-to-user information elements within CCITT Q.931 call control messages, during call establishment and clearing phases. The user-to-user signalling feature can be used for signalling application-specific information among end-stations. For more details, refer to CCITT Recommendation Q.931 (1993), Section 7.

11.5 P channel call control procedures

The IEEE 802.9 P channel provides IEEE 802/ISO 8802 LAN (and possibly additional) packet-mode bearer services. The IEEE 802.9 HMUX sublayer delivers an isochronous clear channel for transport of the P channel information. The size of the P channel is an integer multiple of 64 kb/s TDM-based slots. The bandwidth provisioned for the P channels is normally independent of CCITT Q.931 and is affected by the C channel bandwidth allocation. If explicit bandwidth reservation is required for the P channel, then the following procedures can be applied to establish, release, and dynamically adjust the bandwidth.

11.5.1 P channel establishment and release

When an ISTE determines (typically as part of its initialization procedures) that P channel establishment between itself and the AU is required, the ISTE shall transmit a CCITT Q.931 SETUP message with an extended bearer capability information element. This information element shall have the transfer mode field encoded as "IEEE 802.9 packet mode." This identifies the SETUP message to the AU as a request for the establishment of a P channel connection if none is currently active (otherwise this request is treated as an error; not more than one P channel call may exist within the interface).

The AU, in the absence of errors, shall respond to the ISTE with either a CALL PROCEEDING message, followed by an ALERTING or a CONNECT message. Standard CCITT Q.931 procedures are applied for call establishment and call release. Note that a minimum of sixteen 64 kb/s TDM channels shall be allocated to the P channel in order to meet IEEE 802 functional requirements for bandwidth.

11.5.2 P Channel bandwidth adjustment

In order to enhance bandwidth utilization, the initial allocation of 64 kb/s TDM channels to the P channel may require periodic adjustment, for example, as a result of bandwidth contention between the P channel and C channel(s).

The AU initiates the procedure to adjust the P channel bandwidth. Adjustment to the P channel bandwidth is accomplished using the HOLD and RETRIEVE functions described in CCITT Recommendation Q.932 (1993), Section 6.2. The procedure is as follows:

a) The AU transmits a HOLD message to the ISTE, specifying the call reference assigned to the P channel call.

b) The ISTE, upon receipt of the HOLD message, responds by acquiescing the P channel to/from the AU (e.g., suspends request/grant timing) and transmits a HOLD ACKNOWLEDGE message to the AU.

c) The AU, upon receipt of the ISTE's HOLD ACKNOWLEDGE message, acquiescing the P channel (e.g., withholds pending MAC frame transmissions and grants to the ISTE), reconfigures its local HMUX sublayer to effect the P channel adjustments, and transmits to the ISTE a RETRIEVE message, including an extended channel identification information element to convey the newly readjusted P channel allocations.

d) The ISTE, upon receipt of the RETRIEVE message, reconfigures its local HMUX sublayer to match the allocation described in the RETRIEVE message, transmits a RETRIEVE ACKNOWLEDGE message to the AU, and reenables its P channel.

e) The AU, upon receipt of the RETRIEVE ACKNOWLEDGE message, reenables its P channel (i.e., resumes issuing grants and transmits any queued MAC frames). This completes the P channel bandwidth adjustment procedures.

11.5.3 P and C channel signalling example

An example of the signalling message flow associated with the establishment and management of P and C channel services on the Mode 3 (full bandwidth management) ISLAN interface is described below and is depicted in figure 11-10. This figure is a simplified diagram and does not show all of the necessary messages exchanged during call establishment.

Step 1: ISTE-1 transmits a P channel SETUP message to the AU.

Step 2: The AU initializes its P channel components associated with the requesting ISTE's port and transmits a CONNECT message to the ISTE, including the extended channel identification information element.

Step 3: ISTE-1 initializes its P channel components based on the received channel information. The P channel between ISTE-1 and the AU is now available for use.

NOTE—In the case of the template Modes 0 and 2 (defined in 9.2.3.2), the default P channel is available when the physical connection between the ISTE and the AU is established.

Steps 4–6: The sequence shown in steps 1-3 is repeated between ISTE-2 and the AU. The P channel between ISTE-2 and the AU is then available for use, and by definition of MAC service, a P channel between ISTE-1 and ISTE-2 is now available for use.

Step 7: MAC frames flow through the AU between ISTE-1 and ISTE-2.

Depending upon the implementation, the P channel between ISTE-1 and the AU may be established at the same or different speeds.

In this example, it is assumed that:

a) ISTE-1 is attached to the AU at 4.096 Mb/s and is allocated a P channel of 50×64 kb/s (3.2 Mb/s).

b) ISTE-2 is attached to the AU at 20.48 Mb/s, and is allocated a P channel of 250×64 kb/s (16.0 Mb/s).

This example would require a buffered AU model; non-buffered AU models which require symmetric P channels are also possible.

Step 8: ISTE-1, wishing to establish an isochronous channel to ISTE-2 for a videophone call, transmits a C channel SETUP message to the AU, specifying ISTE-2 as the destination, and provides an H11 bandwidth (24×64 kb/s).

Step 9: The AU, in turn, transmits a C channel SETUP message to ISTE-2 that includes the extended channel identification information element, selecting 24 available time slots.

Step 10: ISTE-2 processes the SETUP message. This processing includes the reconfiguration of the HMUX sublayer to allocate the specified 24 time slots to the SAP associated with its videophone. Having initialized its videophone components, ISTE-2 turns on ringing, and returns an ALERTING message to the AU.

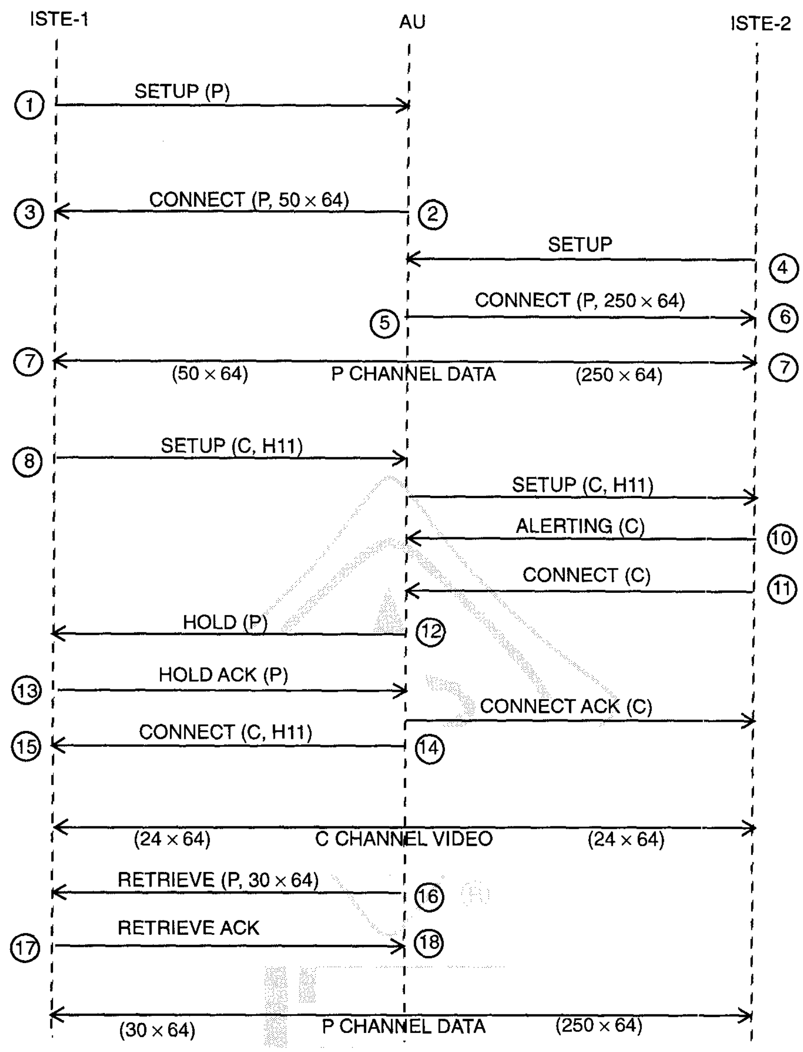


Figure 11-10—P and C channel signalling example

Step 11: The user of the ISTE-2 accepts the videophone call ("goes off-hook"), and ISTE-2 responds by sending a CONNECT message to the AU.

Step 12: The AU determines that there is insufficient bandwidth available in the ISTE-1 interface for the requested H11 video call, and begins bandwidth adjustment in accordance with 11.5.2, transmitting a HOLD message, which specifies the P channel call reference, to ISTE-1.

Step 13: ISTE-1 responds by withholding P channel requests, temporarily releasing the P channel's time slots, and returning a HOLD ACKNOWLEDGE message to the AU.

Step 14: The AU completes the videophone call by transmitting a CONNECT message, which specifies the C channel call reference, and the extended channel identification information element, which specifies 24 available time slots (some of which were taken from the P channel call which is now on hold).

Step 15: ISTE-1 processes the CONNECT message, which includes an adjustment to its HMUX sublayer to allocate the specified 24 available time slots to the SAP associated with its videophone. The videophone call is now complete end-to-end.

Step 16: The AU transmits a RETRIEVE message to ISTE-1, which specifies the P channel call reference, and includes the extended channel identification information element that reduces the previous 50 × 64 kb/s (3.2 Mb/s) down to 30 × 64 kb/s (1.92 Mb/s) in order to accommodate the time slot now in use by the H11 videophone call.

Step 17: ISTE-1 reconfigures its HMUX sublayer to the adjusted P channel map, and returns a RETRIEVE ACKNOWLEDGE message to the AU.

Step 18: The AU processes the RETRIEVE ACKNOWLEDGE by reenabling the P channel to ISTE-1, which can now resume MAC frame transmission and reception at the new rate.

11.6 Interface between CCITT Q.931 and the HMC

The highest sublayer within the PHY Layer is the HMUX sublayer. It is responsible for processing the contents of the hybrid multiplexer control (HMC) field (refer to 6.5.5 and 9.2) in the IEEE 802.9 TDM frame, and interfaces both to the IEEE 802.9 MAC sublayer and to the IEEE 802.9 isochronous services. The functions of the HMUX sublayer are to:

a) Multiplex the streams of information from the higher layers into a single stream of octets for acceptance by the PS sublayer

b) Demultiplex the stream of octets received by the PS sublayer into a set of channels each directed toward an appropriate higher layer SAP

c) Provide the functionality needed to dynamically modify the allocation of TDM slots in the payload field of the TDM frame in order to enable the connection and disconnection of individual N × 64 kb/s channels without disrupting the activity of other service channels

Figure 11-11 illustrates the coordinated role that the signalling protocol exchange and the HMC protocol exchange provide in the function of the dynamic bandwidth allocation and reallocation mechanism.

At time of initialization, the signalling and ISDN management module will communicate to the HMUX sublayer's LME the parameters that are necessary to configure the HMUX. Among the parameters will be the location and size of the various service channels that constitute the IEEE 802.9 TDM frame. At this time the HMC control module will be instructed to "load" the bandwidth assignment template into the foreground template register. The HMC control module will be also instructed as to the bearer service mode (template identification Modes 0, 1, 2, or 3).

Once the PS sublayer has become synchronized with the peer sublayer (if this is the ISTE, then the peer would be the AU), then the HMC control module would establish HMUX multiframe operation with the peer HMC. The HMUX multiframe operation provides an insurance against misinterpretation of the received HMC field. Since the HMC field provides the realtime synchronization signal (via the template exchange notification bit), it is critical that this information be protected from corruption. Once the HMC control module has reached stable operation with its peer, the HMUX sublayer is able to function to pass user information across the IEEE 802.9 subnetwork.

For the case of an ISTE, when the signalling and ISDN management module receives an application request, it will exchange a CCITT Q.931 message to request that a connection be established or released for the amount of bandwidth that the application requires. These CCITT Q.931 messages are sent across the D channel to the AU. The information elements contain the requested size of channel information and the identity of the TDM slots that constitute this information stream. Once the AU confirms the request via a CCITT Q.931 message, the signalling and ISDN management module passes this information down to the

HMUX sublayer's LME. In response, the HMC control module loads the "new" information into the background template register. After that is complete, the HMC control module exercises the procedures for exchange of the template exchange notification bit so that the remote peer (in the AU) can simultaneously effect a swap from the background template register to the foreground template register. After the exchange hysteresis requirements have been satisfied, the HMUX sublayers in the ISTE and the AU would be operating in correct synchronization.

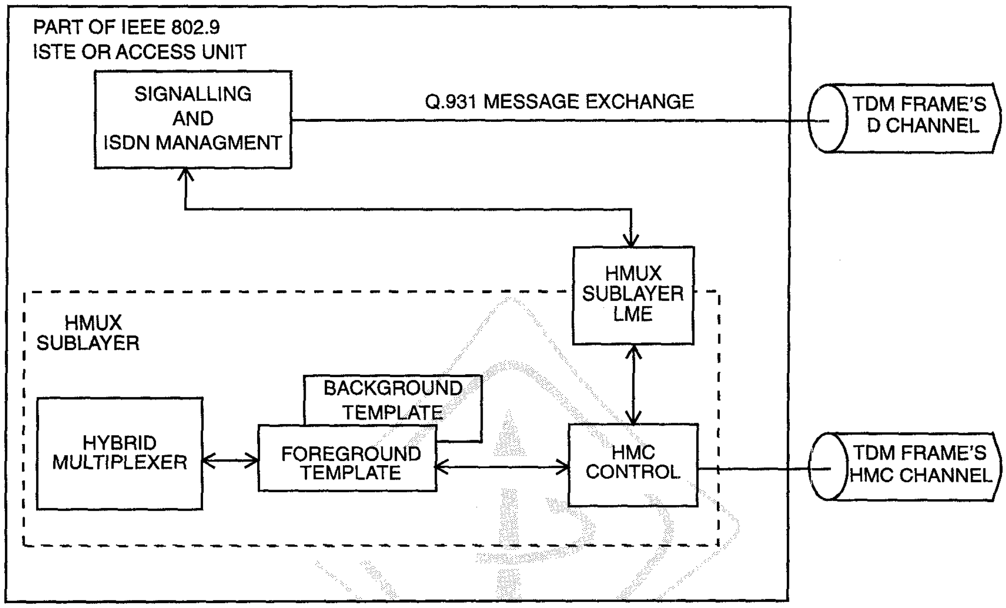


Figure 11-11—Coordination between HMC and signalling

11.7 Addressing

For the conveyance of the called and the calling party number/subaddress, CCITT Q.931 defines information elements, as depicted in figure 11-5. These information elements can be used to support different addressing schemes, namely, IEEE 802 MAC, CCITT E.164 (ISDN), those conforming to ISO Network Layer addressing schemes (ISO/IEC 8348:1993 Addendum 2), and other addressing techniques such as internet protocol (IP) and internet protocol X (IPX). The type/numbering plan identification fields within these information elements can be used to specify the addressing scheme that is currently being used. Annex H describes in detail which addressing schemes can be used for various network topologies.

11.8 Configuration topologies

The ISTE negotiates with, or through, an AU for the management (establishment and control) of isochronous communication channels between themselves and:

— Other ISTEs in the IEEE 802.9 LAN
— Other SAPs in WANs

The communication channels managed in this manner include:

— Circuit-mode B and C channels
— Packet-mode B, C, and D channels

Depending upon the AU implementation, configuration topology, and other considerations, the management of these communications channels may be provided by the AU, a PBX behind the AU, an ISDN behind the AU PBX, or by combinations of these elements. In any of these configurations it is recommended that the CCITT Recommendation Q.931 (1993) procedures, conveyed in the D channel of the IEEE 802.9 ISTE interface, provide the basis for management and provisioning of the communication channels. The procedures of this interface should be independent of specific ISTE, AU, and ISDN network implementations, and thus contribute to interoperability across the interface. The AU implementation is a matter for future study.

Figure 11-12 presents an overview of the types of communication pathways over which ISDN management (signalling) information and LAN station management information may be exchanged between networks and the AU. From the AU, this management information may need to be "transformed" into that information which will be destined toward the ISTE. Note that in the more general situation, this terminal equipment (TE) will consist of terminal adaptors as well as native mode IEEE 802.9 terminals.

11.9 ISDN management functions

CCITT Recommendation Q.931 (1993) has been designed to operate as a full "management" protocol as well as to provide circuit provisioning. As such, there is the opportunity to conduct the same range of functional management exchange that is commonly associated with simple network management protocol (SNMP) LAN station management.

These management functions may be classified according to the following fields of applications for:

a) Maintenance
   1) Fault tracing
   2) Spontaneous error reporting
   3) Error threshold alarm reporting
   4) Continuous monitoring
   5) Diagnostic testing
   6) Resource (re)initialization
   7) Confidence testing
   8) Resource identification
   9) Trouble isolation
   10) Audit
b) Configuration management
   1) Routing changes
   2) Database changes
   3) Equipment identification
c) Accounting management (billing data reporting)
d) Performance management
   1) Traffic data reporting
   2) Performance monitoring
   3) Applying controls
e) Security management

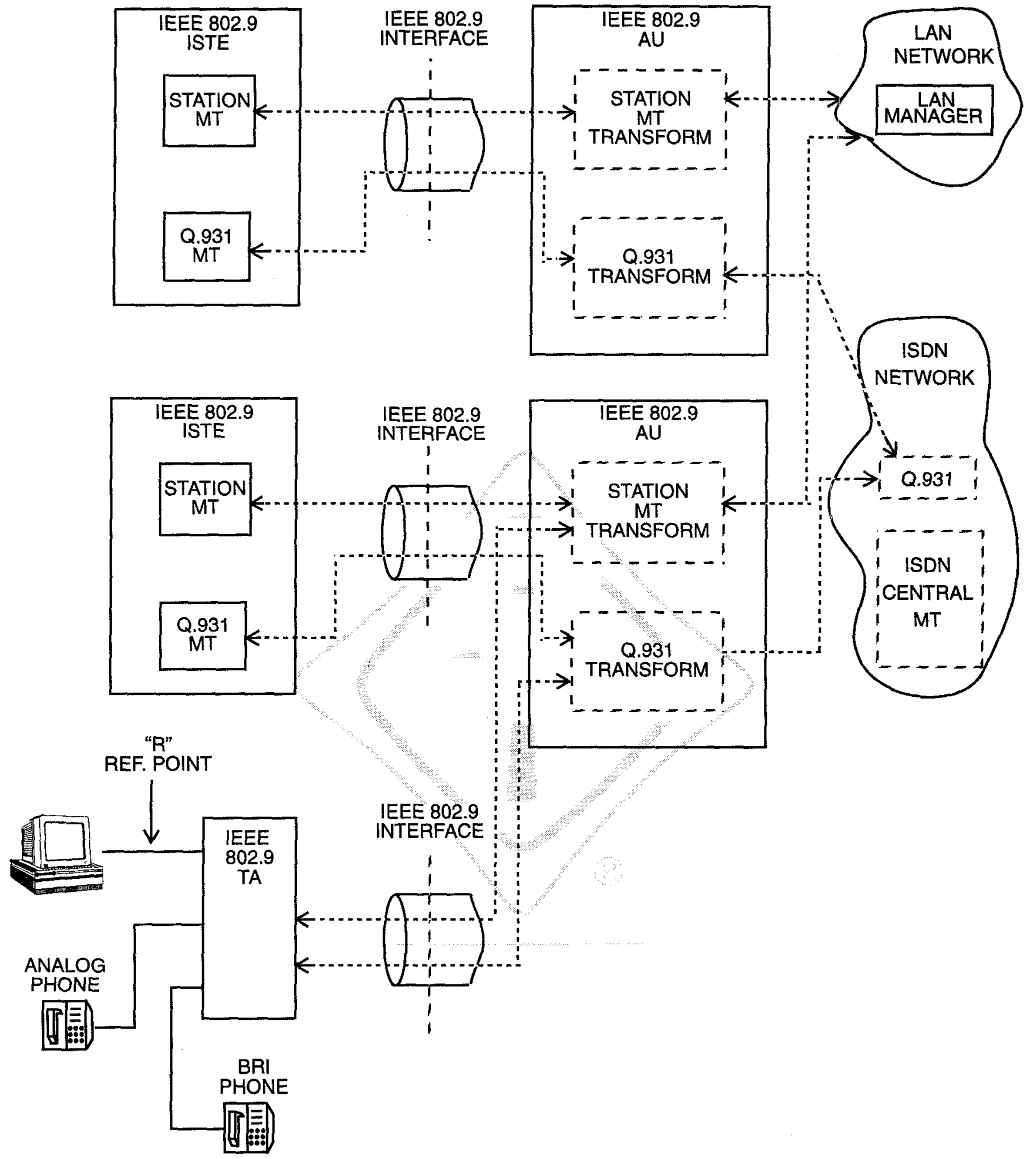


Figure 11-12—Management transport across IEEE 802.9 network

Annex A
Protocol implementation conformance statement (PICS) proforma

(normative)

This annex is a matter for further study.

Annex B
Managed object conformance statement (MOCS)

(normative)

This annex is a matter for further study.

Annex C
GDMO and ASN.1 specification of management information

(normative)

C.1 Introduction

This annex contains Guidelines for the Definition of Managed Objects (GDMO) specifications of information elements required for the full definition of ISLAN systems and layer management and, as such, this annex is an integral, normative part of this standard.

C.2 Specification for the MAC service access point managed object

The following specifications define the information elements of and related to the MAC service access point (MSAP) managed object (MO).

C.2.1 Attributes of the MSAP MO

The attribute, mac-address, is the universally or locally administered 48-bit address.

```
mac-address  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX MSAP-Attribute-Module.MACAddrBitString;
    MATCHES FOR  Equality;
    BEHAVIOR  mac-addr-behavior;

    REGISTERED AS  { islan mac(0) attribute(7) mac-address(0) };
```

The attribute, mac-service-user, identifies the MAC service user (e.g., an LLC entity) that is associated with the MSAP being characterized.

```
mac-service-user  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX MSAP-Attribute-Module.MACServiceUserDesc;

    MATCHES FOR  Equality;
    BEHAVIOR  mac-serv-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mac-service-user(1) };
```

C.2.2 Name binding of the MSAP MO

The name binding, mac-name, allows the layer management entity (LME) and the system management application process (SMAP) to distinguish MSAP MOs. It also defines the containment relationship between an MSAP MO and the generic MAC entity MO that contains it. It defines the creation and deletion operations for the MSAP MO (but not the MSAP itself).

```
msap-name  NAME BINDING

    SUBORDINATE OBJECT CLASS  msap;
    NAMED BY
      SUPERIOR OBJECT CLASS  generic-mac-entity
    WITH ATTRIBUTE  mac-address;
```

```
    BEHAVIOR  msap-name-behavior;
    CREATE   mac-addr;
    DELETE   only-if-no-contained-objects;

    REGISTERED AS { islan mac(0) nameBinding(6) msap-name(0) };
```

The parameter, mac-addr, contains the MAC address supplied with a create operation for an MSAP MO.

```
mac-addr   PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX   MSAP-Attribute-Module.MACAddrBitString;

    REGISTERED AS {islan mac(0) parameter(5) mac-addr(0)};
```

The MSAP MO has two attributes: the MAC address of the SAP and the Data Link Layer (DL) user that exchanges MAC service data units (MSDUs) across that SAP. The MSAP MO is named by the distinguished name of its containing superior MO class, the mac-entity. Thus, there is a mechanism for distinguishing MSAP MOs.

C.2.3 Associated ASN.1 specifications

The MSAP MO specification requires the following Abstract Syntax Notation One (ASN.1):

```
MSAP-Attribute-Module { islan mac(0) asn1Module(2) msap-asn-1(0) }

    IMPORTS MAC-Entity-Id FROM Generic-MAC-Attribute-Module;
    DEFINITIONS IMPLICIT TAGS ::=

      BEGIN
        MACAddrBitString::= BIT STRING { first(0),last(47) }
        MACServiceUserType::= ENUMERATED { mac-entity(0),
                                           lapd(1),lapb(2) }
        MACServiceUserDesc::=SEQUENCE
                                {
                                 macServiceUser[0]MACServiceUserType
                                 macName       [1]MAC-Entity-Id
                                }
         END
```

C.3 Specifications for the generic MAC entity MO

The following specifications define the information elements relevant to the generic MAC entity MO, as referenced in the GDMO in 10.10.1.2 of this standard.

C.3.1 Attributes of the generic MAC entity MO

The attribute, mac-entity-id, is a naming attribute for the generic MAC entity MO.

```
mac-entity-id   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.MAC-Entity-Id;
    MATCHES FOR Equality;
    BEHAVIOR mac-entity-id-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mac-entity-id(2) };
```

The attribute, manufacturer-id, identifies the manufacturer of the implementation represented by the MAC entity MO.

```
manufacturer-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "IEEE Std 802.1F-1993":ManufacturerID;
    BEHAVIOR manufacturer-id-behavior;

    REGISTERED AS { islan mac(0) attribute(7) manufacturer-id(3) };
```

The attribute, manufacturer-product-id, identifies the product implementation represented by the MAC entity MO.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "IEEE Std 802.1F-1993": ManufacturerProductID;
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan mac(0) attribute(7) manufacturer-product-
            id(4) };
```

The attribute, manufacturer-product-version, identifies the product implementation model represented by the MAC entity MO.

```
manufacturer-product-version  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "IEEE Std 802.1F-1993":
            ManufacturerProductVersion;
    BEHAVIOR manufacturer-product-version-behavior;

    REGISTERED AS { islan mac(0) attribute(7) manufacturer-product-
            version(5) };
```

The attribute, max-mpdu-size, contains the size of the largest MAC protocol data unit (MPDU) that can be received.

```
max-mpdu-size  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.MPDU-Size;
    MATCHES FOR Equality;
    BEHAVIOR max-mpdu-size-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-mpdu-size(6) };
```

The attribute, max-mpdu-size-received, contains the size of the largest MPDU that has been received.

```
max-mpdu-size-received  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR max-mpdu-size-received-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-mpdu-size-received(7) };
```

The attribute, max-priority-requested, contains the highest priority request so far received, in an MA-UNITDATA request.

```
max-priority-requested  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
```

```
    MATCHES FOR Equality;
    BEHAVIOR max-priority-requested-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-priority-requested(8) };
```

The attribute, min-mpdu-size, contains the size of the smallest MPDU that can be received.

```
min-mpdu-size  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.MPDU-Size;
    MATCHES FOR Equality;
    BEHAVIOR min-mpdu-size;

    REGISTERED AS { islan mac(0) attribute(7) min-mpdu-size(9) };
```

The attribute, min-priority-requested, contains the lowest priority request so far received, in an MA-UNITDATA request.

```
min-priority-requested  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR min-priority-requested-behavior;

    REGISTERED AS { islan mac(0) attribute(7) min-priority-
            requested(10) };
```

The attribute, mpdu-octets-received-count, contains the accumulated number of octets received during the reception of MPDUs.

```
mpdu-octets-received-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-octets-received-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-octets-received-
            count(11) };
```

The attribute, mpdu-octets-sent-count, contains the accumulated number of octets sent during the transmission of MPDUs.

```
mpdu-octets-sent-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-octets-sent-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-octets-sent-
            count(12) };
```

The attribute, mpdu-received-count, contains the number of MPDUs received.

```
mpdu-received-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-received-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-received-count(13) };
```

The attribute, mpdu-received-in-error-count, contains the number of MPDUs received that contained errors.

```
mpdu-received-in-error-count   ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":tideMark;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-received-in-error-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-received-in-error-
            count(14) };
```

The attribute, mpdu-received-in-error-thres, is a threshold for the mpdu-received-in-error counter.

```
mpdu-received-in-error-thres   ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992":
            corruptedPDUsReceivedThreshold;
    BEHAVIOR  mpdu-received-in-error-thres-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-received-in-error-
            thres(15) };
```

The attribute, mpdu-sent-count, is a counter of the total number of MPDUs sent.

```
mpdu-sent-count   ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": octetsReceivedCounter;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-sent-count;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-sent-count(16) };
```

The attribute, priority-last-requested, contains the last priority value requested in an MA-UNITDATA request.

```
priority-last-requested   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.Priority;
    MATCHES FOR Equality;
    BEHAVIOR priority-last-requested-BEHAVIOR;

    REGISTERED AS { islan mac(0) attribute(7) priority-last-
            requested(17) };
```

The attribute, qos-profile-last-requested, is the last quality of service (QOS) profile requested by the MAC service user.

```
qos-profile-last-requested   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.QosProfile;
    MATCHES FOR Equality;
    BEHAVIOR qos-profile-last-requested-behavior;

    REGISTERED AS { islan mac(0) attribute(7) qos-profile-last-req(18) };
```

The attribute, resource-type, identifies the IEEE 802 standard applicable to the MAC entity MO characterized.

```
resource-type  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX  "IEEE Std 802.1F-1993":ResourceType;
```

```
    BEHAVIOR  resource-type-behavior;

    REGISTERED AS { islan mac(0) attribute(7) resource-type(19) };
```

The attribute, rx-max-err-count, is a counter of the number of MPDUs received that exceeded the maximum MPDU size limit.

```
rx-max-err-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR Equality;
    BEHAVIOR rx-max-err-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-max-err-count(20) };
```

The attribute, rx-max-err-count-thres, is a counter threshold for the rx-max-err-count attribute.

```
rx-max-err-count-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR Equality;
    BEHAVIOR rx-max-err-count-thres;

    REGISTERED AS { islan mac(0) attribute(7) rx-max-err-count-
            thres(21) };
```

The attribute, rx-min-err-count, is a counter of the number of MPDUs received, the size of which was less than the minimum MPDU size limit.

```
rx-min-err-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR Equality;
    BEHAVIOR rx-min-err-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-min-err-count(22) };
```

The attribute, rx-min-err-count-thres, is a counter threshold for the rx-min-err-count attribute.

```
rx-min-err-count-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR Equality;
    BEHAVIOR rx-min-err-count-thres;

    REGISTERED AS { islan mac(0) attribute(7) rx-min-err-count-
            thres(23) };
```

The attribute, rx-size-last-mpdu, contains the size of the last MPDU received.

```
rx-size-last-mpdu  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-MAC-Attribute-Module.MPDU-Size;
    MATCHES FOR Equality;
    BEHAVIOR rx-size-last-mpdu-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-size-last-mpdu(24) };
```

The attribute, rx-grant-timer-err-count, contains the number of times the grant timer expired while waiting for a grant.

```
rx-grant-timer-err-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
    MATCHES FOR Equality;
    BEHAVIOR rx-grant-timer-err-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-grant-timer-err
            count(25) };
```

The attribute, rx-grant-timer-err-thres, is a counter threshold for rx-grant-timer-err-count attribute.

```
rx-grant-timer-err-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
    MATCHES FOR Equality;
    BEHAVIOR rx-grant-timer-err-thres-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-grant-timer-err-
            thres(26) };
```

The attribute, rx-res-err, identifies that a resource is not available for reception of an MPDU after a grant was issued.

```
rx-res-err  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
    MATCHES FOR Equality;
    BEHAVIOR rx-res-err-behavior;

    REGISTERED AS { islan mac(0) attribute(7) rx-res-err(27) };
```

The attribute, max-request-time, specifies the time for a request to be honored after an MPDU is ready for transmission.

```
max-request-time  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
    MATCHES FOR Equality;
    BEHAVIOR max-request-time-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-request-time(28) };
```

The attribute, max-req-err-count, specifies the number of times the max-request-time has expired.

```
max-req-err-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
    MATCHES FOR Equality;
    BEHAVIOR max-req-err-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-req-err-count(29) };
```

The attribute, max-req-err-count-thres, is a counter threshold for max-req-err-count attribute.

```
max-req-err-count-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992"tideMark;
```

```
    MATCHES FOR Equality;
    BEHAVIOR max-req-err-count-thres-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-req-err-count
             thres(30)};
```

The attribute, standard-revision, identifies the latest revision applicable to the attribute, resource-type.

```
standard-revision  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "IEEE Std 802.1F-1993":StandardRevision;
    BEHAVIOR  standard-revision-behavior;

    REGISTERED AS { islan mac(0) attribute(7) standard-revision(31) };
```

C.3.2 Actions of the generic MAC entity MO

The action, activate-mac, activates a suspended MAC entity.

```
activate-mac  ACTION

    BEHAVIOR                    activate-mac-behavior;
    MODE                        CONFIRMED;
    PARAMETERS                  initial-activation-values;
    WITH INFORMATION SYNTAX     ParmValues;
    WITH REPLYSYNTAX            ActivateResponse;

    REGISTERED AS { islan mac(0) action(9) activate-mac(0) };
```

The action, deactivate-mac, suspends a MAC entity.

```
deactivate-mac  ACTION

    BEHAVIOR      deactivate-mac-behavior;
    MODE          NON-CONFIRMED;

    REGISTERED AS { islan mac(0) action(9) deactivate-mac(1) };
```

The action, reset-mac, resets a MAC entity.

```
reset-mac  ACTION

    BEHAVIOR                    reset-mac-behavior;
    MODE                        NON-CONFIRMED;
    PARAMETERS                  initial-reset-values;
    WITH INFORMATION SYNTAX     ParmValues;

    REGISTERED AS { islan mac(0) action(9) reset-mac(2) };
```

C.3.3 Notification of the generic MAC entity MO

The notification, mpdu-received-in-error-thres-not, notifies the SMAP that the mpdu-received-in-error-thres threshold has been reached.

```
mpdu-received-in-error-thres-not  NOTIFICATION

    BEHAVIOR mpdu-received-in-error-not-behavior;
    MODE NON-CONFIRMED;
```

```
    REGISTERED AS { islan mac(0) notification(10) mpdu-error-not(0) };
```

The notification, rx-max-err-count-thres-not, notifies the SMAP that the rx-max-err-count-thres threshold has been reached.

```
rx-max-err-count-thres-not  NOTIFICATION

    BEHAVIOR rx-max-err-count-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) rx-max-err-count-
            thres-not(1) };
```

The notification, rx-min-err-count-thres-not, notifies the SMAP that the rx-min-err-count-thres threshold has been reached.

```
rx-min-err-count-thres-not  NOTIFICATION

    BEHAVIOR rx-min-err-count-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) rx-min-err-count-
            thres-not(2) };
```

The notification, rx-grant-timer-err-thres-not, notifies the SMAP that the rx-grant-timer-err-thres threshold has been reached.

```
rx-grant-timer-err-thres-not  NOTIFICATION

    BEHAVIOR rx-grant-timer-err-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) rx-grant-timer-err-
            thres-not(3) };
```

The notification, max-req-err-count-thres-not, notifies the SMAP that the max-req-err-count-thres threshold has been reached.

```
max-req-err-count-thres-not  NOTIFICATION

    BEHAVIOR max-req-err-count-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) max-req-err-count-
            thres-not(4) };
```

C.3.4 Name binding of the generic MAC entity MO

The name binding, mac-entity-name, allows the SMAP to distinguish MAC entity MOs. It also defines the containment relationship between a managed system and its MAC entity MO. It defines the create and delete operations for the MAC entity MO (but not the MAC entity itself).

```
mac-entity-name  NAME BINDING
    SUBORDINATE OBJECT CLASS  generic-mac-entity;
    NAMED BY
            SUPERIOR OBJECT CLASS "ISO/IEC 10165-2: 1992":system;
    WITH ATTRIBUTE  mac-entity-id;
    BEHAVIOR       mac-entity-name-behavior
    CREATE         mac-entity-id
```

```
                  manufacturer-id
                  manufacturer-product-id
                  manufacturer-product-version
                  max-mpdu-size
                  max-priority-requested
                  min-mpdu-size
                  min-priority-requested
                  mpdu-octets-received-count
                  mpdu-octets-sent-count
                  mpdu-received-count
                  mpdu-received-in-error-count
                  mpdu-received-in-error-thres
                  mpdu-sent-count
                  priority-last-requested
                  qos-profile-last-requested
                  resource-type
                  rx-max-err-count
                  rx-max-err-count-thres
                  rx-min-err-count
                  rx-min-err-count-thres
                  size-last-mpdu
                  rx-grant-timer-err-count
                  rx-grant-timer-err-thres
                  rx-res-err
                  max-request-time
                  max-req-err-count
                  max-req-err-count-thres
                  standard-revision;
    DELETE        deletes-contained-objects;

    REGISTERED AS { islan mac(0) nameBinding(6) mac-entity-name(1) };
```

NOTE— Refer to tables 10-1 through 10-13 for name binding for mandatory MOs.

The parameter, mac-entity-id, is the MAC MO name supplied with the create operation.

```
mac-entity-id  PARAMETER

    CONTEXT        ACTION-INFO;
    WITH SYNTAX    Generic-MAC-Attribute-Module.MAC-Entity-Id;

    REGISTERED AS { islan mac(0) parameter(5) mac-entity-id(1) };
```

The parameter, manufacturer-id, is supplied with the create operation as the manufacturer-id.

```
manufacturer-id  PARAMETER

    CONTEXT        ACTION-INFO;
    WITH SYNTAX    "IEEE Std 802.1F-1993":ManufacturerID;

    REGISTERED AS { islan mac(0) parameter(5) manufacturer-id(2) };
```

The parameter, manufacturer-product-id, is supplied with the create operation as the manufacturer-product-id.

```
manufacturer-product-id  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX "IEEE Std 802.1F-1993":ManufacturerProductID;

    REGISTERED AS { islan mac(0) parameter(5) manufacturer-product-
            id(3) };
```

The parameter, manufacturer-product-version, is supplied with the create operation as the manufacturer-product-version.

```
manufacturer-product-version   PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX "IEEE Std 802.1F-1993":ManufacturerProductVersion;

    REGISTERED AS { islan mac(0) parameter(5) manufacturer-product-
            version(4) };
```

The parameter, max-mpdu-size, is supplied with the create operation as the maximum size of an MPDU for the MAC protocol specification in question.

```
max-mpdu-size   PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX   MaxMpduSize;

    REGISTERED AS { islan mac(0) parameter(5) max-mpdu-size(5) };
```

The parameter, max-priority-requested, is supplied with the create operation with a default value for maximum priority requested for the MAC protocol specification in question. The actual value depends on the user requirements.

```
max-priority-requested   PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX   MaxPriority-Requested;

    REGISTERED AS { islan mac(0) parameter(5) max-priority-requested(6) };
```

The parameter, min-mpdu-size, is supplied with the create operation as the minimum size of an MPDU for the MAC protocol specification in question.

```
min-mpdu-size   PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX   MinMpduSize;

    REGISTERED AS { islan mac(0) parameter(5) min-mpdu-size(7) };
```

The parameter, min-priority-requested, is supplied with the create operation with a default value for minimum priority requested for the MAC protocol specification in question. The actual value depends on the user requirements.

```
min-priority-requested   PARAMETER

    CONTEXT    ACTION-INFO;
    WITH SYNTAX   MinPriorityRequested;
    REGISTERED AS { islan mac(0) parameter(5) min-priority-requested(8) };
```

The parameter, mpdu-octets-received-count, is supplied with the create operation with a default value for mpdu-octets-received-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-octets-received-count   PARAMETER

    CONTEXT   ACTION-INFO;
```

```
    WITH SYNTAX  MpduOctetsReceivedCount;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-octets-received-
            count(9) };
```

The parameter, mpdu-octets-sent-count, is supplied with the create operation with a default value for mpdu-octets-sent-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-octets-sent-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MpduOctetsSentCount;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-octets-sent-
            count(10) };
```

The parameter, mpdu-received-count, is supplied with the create operation with a default value for mpdu-received-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-received-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MpduReceivedCount;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-received-count(11) };
```

The parameter, mpdu-received-in-error-count, is supplied with the create operation with a default value for mpdu-received-in-error-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-received-in-error-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MpduReceivedInErrorCount;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-received-in-error-
            count(12) };
```

The parameter, mpdu-received-in-error-thres, is supplied with the create operation as the threshold value for the attribute, mpdu-received-in-error-count.

```
mpdu-received-in-error-thres  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MpduReceivedInErrorThres;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-received-in-error-
            thres(13) };
```

The parameter, mpdu-sent-count, is supplied with the create operation with a default value for mpdu-sent-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-sent-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MpduSentCount;
```

```
    REGISTERED AS { islan mac(0) parameter(5) mpdu-sent-count(14) };
```

The parameter, priority-last-requested, is supplied with the create operation with a default value for priority-last-requested requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
priority-last-requested  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  PriorityLastRequested;

    REGISTERED AS { islan mac(0) parameter(5) priority-last-
            requested(15) };
```

The parameter, qos-profile-last-requested, is a matter for further study. The parameter, resource-type, is supplied with the create operation as the resource-type.

```
resource-type  PARAMETER

    CONTEXT   ACTION-INFO;
    WITH SYNTAX  "IEEE Std 802.1F-1993":ResourceType

    REGISTERED AS { islan mac(0) parameter(5) resource-type(16) };
```

The parameter, rx-max-err-count, is supplied with the create operation with a default value for rx-max-err-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
rx-max-err-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  RxMaxErrCount

    REGISTERED AS { islan mac(0) parameter(5) rx-max-err-count(17) };
```

The parameter, rx-max-err-count-thres, is supplied with the create operation as the threshold value for the attribute, rx-max-err-count.

```
rx-max-err-count-thres  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  RxMaxErrCountThres;

    REGISTERED AS { islan mac(0) parameter(5) rx-max-err-count-
            thres(18) };
```

The parameter, rx-min-err-count, is supplied with the create operation with a default value for rx-min-err-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
rx-min-err-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  RxMinErrCount;

    REGISTERED AS { islan mac(0) parameter(5) rx-min-err-count(19) };
```

The parameter, rx-min-err-count-thres, is supplied with the create operation as the threshold value for the attribute, rx-min-err-count.

```
rx-min-err-count-thres   PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  RxMinErrCount;

    REGISTERED AS { islan mac(0) parameter(5) rx-min-err-count-
             thres(20) };
```

The parameter, size-last-mpdu is supplied with the create operation with a default value for size-last-mpdu requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
size-last-mpdu   PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  SizeLastMpdu;

    REGISTERED AS { islan mac(0) parameter(5) size-last-mpdu(21) };
```

The parameter, req-grant-timer-err-count, is supplied with the create operation with a default value for reqgrant-timer-err-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
req-grant-timer-err-count   PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  ReqGrantTimerErrorCount;

    REGISTERED AS { islan mac(0) parameter(5) req-grant-timer-error-
             count(22) };
```

The parameter, req-grant-timer-err-thres, is supplied with the create operation with a default value for req-grant-timer-err-thres requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
req-grant-timer-err-thres   PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  ReqGrantTimerErrorThres;

    REGISTERED AS { islan mac(0) parameter(5) req-grant-timer-error-
             thres(23) };
```

The parameter, rx-res-err, is supplied with the create operation with a default value for rx-res-err requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
rx-res-err   PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  RxResErr;

    REGISTERED AS { islan mac(0) parameter(5) rx-res-err(24) };
```

The parameter, max-req-time, is supplied with the create operation with a default value for max-req-time requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
max-req-time  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MaxReqTime;
    REGISTERED AS { islan mac(0) parameter(5) max-req-time(25) };
```

The parameter, max-req-err-count, is supplied with the create operation with a default value for max-req-err-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
max-req-err-count  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MaxReqErrCount;

    REGISTERED AS { islan mac(0) parameter(5) max-req-err-count(26) };
```

The parameter, max-req-err-count-thres, is supplied with the create operation with a default value for max-req-err-count-thres requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
max-req-err-count-thres  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  MaxReqErrCountThres;

    REGISTERED AS { islan mac(0) parameter(5) max-req-err-count-
            thres(27) };
```

The parameter, stand-rev, is supplied with the create operation as the standard-revision.

```
stand-rev  PARAMETER

    CONTEXT  ACTION-INFO;
    WITH SYNTAX  "IEEE Std 802.1F-1993":StandardRevision;

    REGISTERED AS { islan mac(0) parameter(5) stand-rev(28) };
```

C.3.5 Associated ASN.1 specifications

The specification of the generic MAC entity MO requires the following ASN.1:

```
Generic-MAC-Attribute-Module { islan mac(0) asn1Module(2) generic-mac-
            asn-1 (1) }
  DEFINITIONS IMPLICIT TAGS ::=

    BEGIN
    IslanMacTypes::=  ENUMERATED:{ iste(0), au(1) }
    MAC-Entity-Id::=  GraphicString
    MPDU-Size      ::=  INTEGER
    Priority       ::=  INTEGER;
    QosProfile     ::=  SEQUENCE
                         {
                          transitDelay       [0] REAL, -- microseconds
                          residualErrorRate  [1] REAL,
                          probOfLostInfo     [2] REAL
                         }
    END
```

C.4 Specifications for the core ISLAN MAC package

The following specifications define the information elements relevant to the core-islan-mac-characteristics package as referenced in the GDMO in 10.13.1.3.1.1 of this standard.

C.4.1 Attributes of the core-islan-mac-characteristics package

The attribute, islan-mac-type, distinguishes an ISLAN MAC entity MO as belonging either to an access unit (AU) or an ISTE.

```
islan-mac-type  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.ISLAN-MAC-Type;
    MATCHES FOR Equality;
    BEHAVIOR islan-mac-type-behavior;

    REGISTERED AS { islan mac(0) attribute(7) islan-mac-type(32) };
```

The attribute, max-grant-time, contains a value that specifies the maximum waiting time for a GRANT=YES to be sent.

```
max-grant-time  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Time-Value;
    MATCHES FOR Equality;
    BEHAVIOR max-grant-time-behavior;

    REGISTERED AS { islan mac(0) attribute(7) max-grant-time(33) };
```

The attribute, mpdu-to-send, contains a value indicating whether the MAC entity has an MPDU ready to send.

```
mpdu-to-send  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;
    MATCHES FOR Equality;
    BEHAVIOR mpdu-to-send-behavior;

    REGISTERED AS { islan mac(0) attribute(7) mpdu-to-send(34) };
```

The attribute, hmux-mac-address-enable, contains a value 1 (disable mode) while waiting for enable indication for provision of MAC services.

```
hmux-mac-address-enable  ATTRIBUTE
      WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;
      MATCHES FOR Equality;
      BEHAVIOR hmux-mac-address-enable-behavior;

      REGISTERED AS { islan mac(0) attribute(7) hmux-mac-address-
             enable(35) };
```

The attribute, rx-grant-time, contains the maximum waiting for a GRANT=YES, while sending REQUEST=YES.

```
rx-grant-time  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Time-Value;
    MATCHES FOR Equality;
```

```
    BEHAVIOR rx-grant-time-behavior;
    REGISTERED AS { islan mac(0) attribute(7) max-grant-time(36) };
```

C.5 Specifications for the ISTE MAC specific package

The following specifications define the information elements relevant to the iste-mac-characteristics package as referenced in the GDMO in 10.13.1.3.1.2 of this standard.

C.5.1 Attributes of the iste-mac-characteristics package

The attribute, iste-aged-mpdu-discard-count, is a counter of the number of MPDUs that have been discarded in the ISTE MAC entity because of aging in the MPDU ready-to-send queue.

```
iste-aged-mpdu-discard-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR Equality;
    BEHAVIOR iste-aged-mpdu-discard-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) iste-aged-mpdu-discard-
            count(37) };
```

The attribute, iste-aged-mpdu-discard-thres, is the threshold for the iste-aged-mpdu-discard-count counter.

```
iste-aged-mpdu-discard-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": threshold-Counter;
    MATCHES FOR Equality;
    BEHAVIOR iste-aged-mpdu-discard-thres-behavior;

    REGISTERED AS { islan mac(0) attribute(7) iste-aged-mpdu-discard-
            thres(38) };
```

The attribute, iste-max-send-mpdu-q-age, contains a time value that is the maximum time that an MPDU may wait in an ISTE queue for transmission across the Physical (PHY) service boundary.

```
iste-max-send-mpdu-q-age  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Time_Value;
    MATCHES FOR Equality;
    BEHAVIOR iste-max-send-mpdu-q-age-behavior;

    REGISTERED AS { islan mac(0) attribute(7) iste-max-send-mpdu-q-
            age(39) };
```

The attribute, iste-ignore-grant, indicates whether permission is given to the MAC entity to send an MPDU without waiting for a GRANT=YES.

```
iste-ignore-grant  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;
    MATCHES FOR Equality;
    BEHAVIOR ignore-grant-behavior;

    REGISTERED AS { islan mac(0) attribute(7) ignore-grant(40) };
```

C.5.2 Notification of the iste-mac-characteristics package

The notification, iste-aged-mpdu-discard-not, notifies the SMAP that the iste-aged-mpdu-discard-thres threshold has been reached.

```
iste-aged-mpdu-discard-not  NOTIFICATION

    BEHAVIOR aged-mpdu-discard-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) iste-aged-mpdu-discard-
            not(3) };
```

C.6 Specifications for the AU MAC specific package

The following specifications define the information elements relevant to the au-mac-characteristics package as referenced in the GDMO in 10.13.1.3.1.3 of this standard.

C.6.1 Attributes of the au-mac-characteristics package

The attribute, au-aged-mpdu-discard-count, is a counter of the number of MPDUs that have been discarded in the AU MAC entity because of aging in the MPDU ready-to-send queue.

```
au-aged-mpdu-discard-count  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR Equality;
    BEHAVIOR au-aged-mpdu-discard-count-behavior;

    REGISTERED AS { islan mac(0) attribute(7) au-aged-mpdu-discard-
            count(41) };
```

The attribute, au-aged-mpdu-discard-thres, is the threshold for the au-aged-mpdu-discard-count counter.

```
au-aged-mpdu-discard-thres  ATTRIBUTE

    DERIVED FROM "ISO/IEC 10165-2: 1992": threshold-Counter;
    MATCHES FOR Equality;
    BEHAVIOR au-aged-mpdu-discard-thres-behavior;

    REGISTERED AS { islan mac(0) attribute(7) au-aged-mpdu-discard-
            thres(42) };
```

The attribute, au-max-send-mpdu-q-age, contains a time value that is the maximum time that an MPDU may wait in an AU queue for transmission across the PHY service boundary.

```
au-max-send-mpdu-q-age  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Time_Value;
    MATCHES FOR Equality;
    BEHAVIOR au-max-send-mpdu-q-age-behavior;

    REGISTERED AS { islan mac(0) attribute(7) au-max-send-mpdu-q-
            age(43) };
```

The attribute, au-ignore-grant, indicates whether permission is given to the MAC entity to send an MPDU without waiting for a GRANT=YES.

```
au-ignore-grant  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;
    MATCHES FOR Equality;
    BEHAVIOR ignore-grant-behavior;

    REGISTERED AS { islan mac(0) attribute(7) au-ignore-grant(44) };
```

The attribute, nb-trans-peers, contains a numerical value that expresses the number of ISTEs connected to an AU.

```
nb-trans-peers  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX ISLAN-MAC-Attribute-Module.Peers;
    MATCHES FOR Equality;
    BEHAVIOR nb-trans-peers-behavior;

    REGISTERED AS { islan mac(0) attribute(7) nb-trans-peers(45) };
```

C.6.2 Notification of the au-mac-characteristics package

The notification, au-aged-mpdu-discard-not, notifies the SMAP that the au-aged-mpdu-discard-thres threshold has been reached.

```
au-aged-mpdu-discard-not  NOTIFICATION

    BEHAVIOR au-aged-mpdu-discard-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan mac(0) notification(10) au-aged-mpdu-discard-
            not(4) };
```

C.6.3 Name binding of the ISLAN MAC entity MO

The name binding, islan-mac-entity name, distinguishes, for the SMAP, different ISLAN MAC entity MOs. It also defines the containment relationship between the ISLAN MAC entity MO and its managed system, as well as the create and delete operations for the MOs.

```
islan-mac-entity-name  NAME BINDING
    SUBORDINATE OBJECT CLASS  islan-mac-entity;
    NAMED BY
      SUPERIOR OBJECT CLASS  "ISO/IEC 10165-2: 1992":system;
    WITH ATTRIBUTE   islan-mac-entity-id;
    BEHAVIOR  islan-mac-entity-name-behavior;
    CREATE   max-grant-time
             mpdu-to-send
             hmux-mac-address-enable
             rx-grant-time
             iste-aged-mpdu-discard-count
             iste-aged-mpdu-discard-thres
             iste-max-send-mpdu-q-age
             iste-ignore-grant
             islan-mac-id
             islan-mac-type
             au-aged-mpdu-discard-count
             au-aged-mpdu-discard-thres
```

```
            au-max-send-mpdu-q-age
            au-ignore-grant
            nb-trans-peers
    DELETE  deletes-contained-objects;
    REGISTERED AS { islan mac(0) nameBinding(6) islan-mac-type(2) };
```

NOTE— Refer to Chapter 10 for name binding for mandatory MOs.

The parameter, max-grant-time, is supplied with the create procedure as the value expressing the maximum time that a GRANT=YES will be asserted without receiving an MPDU from the transmission peer.

```
max-grant-time  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Time-Value;

    REGISTERED AS { islan mac(0) parameter(5) max-grant-time(29) };
```

The parameter, mpdu-to-send, is supplied with the create procedure with a default value for mpdu-to-send requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
mpdu-to-send  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) mpdu-to-send(30) };
```

The parameter, hmux-mac-address-enable, is supplied with the create procedure with a default value for hmux-mac-address-enable requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
hmux-mac-address-enable  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) hmux-mac-address-
            enable(31) };
```

The parameter, rx-grant-time, is supplied with the create procedure with a default value for rx-grant-time requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
rx-grant-time  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) rx-grant-time(32) };
```

The parameter, iste-aged-mpdu-discard-count, is supplied with the create procedure with a default value for iste-aged-mpdu-discard-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
iste-aged-mpdu-discard-count  PARAMETER

    CONTEXT ACTION-INFO;
```

```
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) iste-aged-mpdu-discard-
            count(33) };
```

The parameter, iste-aged-mpdu-discard-thres, is supplied with the create procedure as the threshold value for the count of MPDUs discarded through the queue aging process.

```
iste-aged-mpdu-discard-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Threshold-Bound;

    REGISTERED AS { islan mac(0) parameter(5) iste-aged-mpdu-discard-
            thres(34) };
```

The parameter, iste-max-send-mpdu-q-age, is supplied with the create procedure with a default value for iste-max-send-mpdu-q-age requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
iste-max-send-mpdu-q-age   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) iste-max-send-mpdu-q-
            age(35) };
```

The parameter, iste-ignore-grant, is supplied with the create procedure with a default value.

```
iste-ignore-grant   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) iste-ignore-grant(36) };
```

The parameter, islan-mac-id, is supplied with the create procedure as the naming identifier value.

```
islan-mac-id   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Generic-MAC-Attribute-Module.MAC-Entity-Id;

    REGISTERED AS { islan mac(0) parameter(5) islan-mac-id(37) };
```

The parameter, islan-mac-type, is supplied with the create procedure as an indication of the type of the new ISLAN MO.

```
islan-mac-type   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.ISLAN-MAC-Types;

    REGISTERED AS { islan mac(0) parameter(5) islan-mac-type(38) };
```

The parameter, au-aged-mpdu-discard-count, is supplied with the create procedure with a default value for au-aged-mpdu-discard-count requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
au-aged-mpdu-discard-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) au-aged-mpdu-discard-
            count(39) };
```

The parameter, au-aged-mpdu-discard-thres, is supplied with the create procedure with a default value for au-aged-mpdu-discard-thres requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
au-aged-mpdu-discard-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) au-aged-mpdu-discard-
            thres(40) };
```

The parameter, au-max-send-mpdu-q-age, is supplied with the create procedure as the value expressing the maximum wait of an MPDU in the ready-to-send queue, before being discarded.

```
au-max-send-mpdu-q-age   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Time-Value;

    REGISTERED AS { islan mac(0) parameter(5) au-max-send-mpdu-q-
            age(41) };
```

The parameter, au-ignore-grant, is supplied with the create procedure with a default value for au-ignore-grant requested for the MAC protocol specification in question. The actual value depends on the MAC protocol operation.

```
au-ignore-grant   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) au-ignore-grant(42) };
```

The parameter, nb-trans-peers, is supplied with the create procedure with a default value for nb-trans-peers requested for the MAC protocol specification in question. The value specifies the number of ISTEs connected to an AU.

```
nb-trans-peers   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISLAN-MAC-Attribute-Module.Alternatives;

    REGISTERED AS { islan mac(0) parameter(5) nb-trans-peers(43) };
```

C.6.4 Associated ASN.1 specifications

The specification of the ISLAN MAC entity MO requires the following ASN.1:

```
ISLAN-MAC-Attribute-Module { islan mac(0) asn1Module(2)
            islan-mac-asn-1 (2) }

    DEFINITIONS IMPLICIT TAGS ::=

     BEGIN
       Alternatives      ::= BOOLEAN
       ISLAN-MAC-Types   ::= ENUMERATED { au(0), iste(1) }
       Peers             ::= INTEGER
       Threshold-Bound   ::= INTEGER
       Time_Value        ::= INTEGER
     END
```

C.7 Specification for the HMUX characteristics package

The following specifications define the information elements relevant to the hmux-characteristics package as referenced in the GDMO in 10.13.2.1 of this standard.

C.7.1 Attributes of the HMUX characteristics package

The attribute, hmux-type-id, contains a set of 8 values or identifiers, one of which shall be agreed to by both the ISTE and the AU so that the appropriate fields in the TDM frame are used by the appropriate HMUX sublayers at both ends of the PHY channel.

```
hmux-type-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.hmux-type-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR hmux-type-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) hmux-type-id(0) };
```

The attribute, manufacturer-id, identifies the manufacturer of the implementation of the HMUX characteristics package.

```
manufacturer-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-id-behavior;
    REGISTERED AS { islan phy(1) attribute(7) manufacturer-id(1) };
```

The attribute, manufacturer-product-id, identifies the product implementation model of the HMUX characteristics package.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product-
            id(2) };
```

The attribute, manufacturer-product-version-id, identifies the product version of the implementation of the HMUX characteristics package.

```
manufacturer-product-version-id   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-version-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-version-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product-
            version-id(3) };
```

The attribute, tx-mode-control, defines the current transmit mode of operation. The range of modes is from 0–7. IEEE 802.9 currently specifies operation for Modes 0–3.

```
tx-mode-control   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992"PHY-Module.tx-mode-
            control;
    MATCHES FOR EQUALITY;
    BEHAVIOR tx-mode-control-behavior;

    REGISTERED AS { islan phy(1) attribute(7) tx-mode-control(4) };
```

The attribute, rx-mode-status, defines the current receive mode of operation. The range of modes is from 0–7. IEEE 802.9 currently specifies operation for Modes 0–3.

```
rx-mode-status   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992"PHY-Module.rx-mode-
            status;
    MATCHES FOR EQUALITY;
    BEHAVIOR rx-mode-status-behavior;

    REGISTERED AS { islan phy(1) attribute(7) rx-mode-status(5) };
```

The attribute, mode, defines the bearer service category that can be supported by the IEEE 802.9 entity. The range of modes is from 0–7. IEEE 802.9 currently specifies operation for Modes 0–3.

```
mode   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992"PHY-Module.mode;
    MATCHES FOR EQUALITY;
    BEHAVIOR mode-behavior;

    REGISTERED AS { islan phy(1) attribute(7) mode(6) };
```

The attribute, mf-error-count, specifies the number of multiframing errors. This value is applicable only for Mode 3 entities.

```
mf-error-count   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992".counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR mf-error-count-behavior;

    REGISTERED AS { islan phy(1) attribute(7) mf-error-count(7) };
```

The attribute, mf-error-thres, is the threshold corresponding to the mf-error-count attribute.

```
mf-error-thres   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992".counter-threshold;
    MATCHES FOR EQUALITY;
    BEHAVIOR mf-error-thres-behavior;

    REGISTERED AS { islan phy(1) attribute(7) mf-error-thres(8) };
```

The attribute, d-chan-rate-control, contains a value that specifies the ISDN D channel rate for the transmitter. The rates specified are 16 kb/s and 64 kb/s.

```
d-chan-rate-control   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-MODULE.d-chan-rate-control;
    MATCHES FOR EQUALITY;
    BEHAVIOR d-chan-rate-control-behavior;

    REGISTERED AS { islan phy(1) attribute(7) d-chan-rate-control(9) };
```

The attribute, d-chan-rate-status, contains a value that specifies the ISDN D channel rate for the receiver. The rates specified are 16 kb/s and 64 kb/s.

```
d-chan-rate-status   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-MODULE.d-chan-rate-status;
    MATCHES FOR EQUALITY;
    BEHAVIOR d-chan-rate-status-behavior;

    REGISTERED AS { islan phy(1) attribute(7) d-chan-rate-status(10) };
```

The attribute, nb-ac-fields, contains a value that specifies the number of access control (AC) fields negotiated in the TDM frame. Permitted values are 1–5.

```
nb-ac-fields   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992".counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR nb-ac-fields-behavior;

    REGISTERED AS { islan phy(1) attribute(7) nb-ac-fields(11) };
```

The attribute, secure-control, contains a value that specifies whether or not secure control is invoked. It is a Boolean: value 0 for no security and value 1 for security invocation.

```
secure-control   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992".counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR secure-control-behavior;

    REGISTERED AS { islan phy(1) attribute(7) secure-control(12) };
```

C.7.2 Action of the HMUX characteristics package

The attribute, set-d-chan-rate-control, changes the value of an associated parameter to specify the D channel rate for transmission.

```
set-d-chan-rate-control  ACTION

    WITH ATTRIBUTE SYNTAX PHY-MODULE.d-chan-rate-control;
    PARAMETERS new-rate;
    BEHAVIOR set-d-chan-rate-control-behavior;

    REGISTERED AS { islan phy(1) attribute(9) set-d-chan-rate-
            control(0) };
```

The attribute, set-d-chan-rate-status, changes the value of an associated parameter to specify the D channel rate for reception.

```
set-d-chan-rate-status  ACTION
    WITH ATTRIBUTE SYNTAX PHY-MODULE.d-chan-rate-status;
    PARAMETERS new-rate;
    BEHAVIOR set-d-chan-rate-status-behavior;

    REGISTERED AS { islan phy(1) attribute(9) set-d-chan-rate-status(1) };
```

C.7.3 Parameters of the HMUX characteristics package

The parameter, new-rate-control, is a parameter of the action set-d-chan-rate-control. It specifies the rate to which the D channel is being changed for transmission.

```
new-rate-control  ACTION

    CONTEXT ACTION-INFO;
    WITH SYNTAX PHY-MODULE.d-chan-rate-control;

    REGISTERED AS { islan phy(1) parameters(5) new-rate-control(0) };
```

The parameter, new-rate-status, is a parameter of the action set-d-chan-rate-status. It specifies the rate to which the D channel is being changed for reception.

```
new-rate-status  ACTION

    CONTEXT ACTION-INFO;
    WITH SYNTAX PHY-MODULE.d-chan-rate-status;

    REGISTERED AS { islan phy(1) parameters(5) new-rate-status(1) };
```

C.7.4 Notifications of the HMUX characteristics package

The notification, mf-error-thres-exceeded, delivers an indication to the SMAP that the threshold corresponding to the attribute, mf-error-count, has been exceeded.

```
mf-error-thres-exceeded  NOTIFICATION

    BEHAVIOR mf-error-thres-exceeded-behavior;

    REGISTERED AS { islan phy(1) notification(10) mf-error-thres-
            exceeded(0) };
```

C.8 Specifications for the physical signalling (PS) characteristics package

The following specifications define the information elements relevant to the ps-characteristics package as referenced in the GDMO in 10.13.2.2 of this standard.

C.8.1 Attributes of the PS characteristics package

The attribute, manufacturer-id, identifies the manufacturer of the implementation of the PS characteristics package.

```
manufacturer-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-id-behavior;
    REGISTERED AS { islan phy(1) attribute(7) manufacturer-id(13) };
```

The attribute, manufacturer-product-id, identifies the product implementation model of the PS characteristics package.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product
            -id(14) };
```

The attribute, manufacturer-product-version-id, identifies the product version of the implementation of the PS characteristics package.

```
manufacturer-product-version-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-version-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-version-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product-
            version-id(15) };
```

The attribute, rem-loopback-status, contains the status value (failure or success) of the latest remote loopback requested by the local agent SMAP.

```
rem-loopback-status  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.Status;
    MATCHES FOR EQUALITY;
    BEHAVIOR rem-loopback-status-behavior;

    REGISTERED AS { islan phy(1) attribute(7) rem-loopback-status(16) };
```

The attribute, rx-tdm-parity-error-counter, is a counter of TDM parity errors detected at the PS sublayer.

```
rx-tdm-parity-error-counter  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR EQUALITY;
```

```
    BEHAVIOR rx-tdm-parity-error-counter-behavior;

    REGISTERED AS { islan phy(1) attribute(7) rx-tdm-parity-error(17) };
```

The attribute, rx-tdm-parity-error-thres, is the threshold associated with rx-tdm-parity-error-counter.

```
rx-tdm-parity-error-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR EQUALITY;
    BEHAVIOR rx-tdm-parity-error-thres-behavior;

    REGISTERED AS { islan phy(1) attribute(7) rx-tdm-parity-error-
            thres(18) };
```

The attribute, synch-loss-counter, is a counter of instances of the loss of synchronization, as detected at the PS sublayer.

```
synch-loss-counter  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR synch-loss-counter-behavior;

    REGISTERED AS { islan phy(1) attribute(7) synch-loss-counter(19) };
```

The attribute, synch-loss-thres, is the threshold associated with synch-loss-counter.

```
synch-loss-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR EQUALITY;
    BEHAVIOR synch-loss-thres-behavior;

    REGISTERED AS { islan phy(1) attribute(7) synch-loss-thres(20) };
```

The attribute, synch-status, specifies the status of the synchronization, as detected at the PS sublayer.

```
synch-status  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.synch-status;
    MATCHES FOR EQUALITY;
    BEHAVIOR synch-status-behavior;

    REGISTERED AS { islan phy(1) attribute(7) synch-status(21) };
```

The attribute, parity-check-fail-counter, is a counter of the number of failures of the test initiated at the PS sublayer of the ability of the transmission peer to detect a locally and artificially induced parity error.

```
parity-check-fail-counter  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR parity-check-fail-counter-behavior;

    REGISTERED AS { islan phy(1) attribute(7) parity-check-fail-
            counter(22) };
```

The attribute, parity-check-fail-thres, is the threshold associated with parity-check-fail-counter.

```
parity-check-fail-thres   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR EQUALITY;
    BEHAVIOR parity-check-fail-thres-behavior;

    REGISTERED AS { islan phy(1) attribute(7) parity-check-fail-
            thres(23) };
```

The attribute, rem-parity-check-status, specifies the status of the parity check.

```
rem-parity-check-status   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR rem-parity-check-status-behavior;
    REGISTERED AS { islan phy(1) attribute(7) rem-parity-check-
            status(24) };
```

C.8.2 Actions of the PS characteristics package

The action, initiate-remote-loopback, enables the local agent SMAP to command the initiation of a remote loopback at the PS sublayer of the local node. The mode is confirmed because management applications, i.e., the managing SMAP, will need to follow this action request with a GET of the local rem-loopback-status attribute.

```
initiate-remote-loopback   ACTION

    BEHAVIOR initiate-remote-loopback-behavior;
    MODE CONFIRMED;

    REGISTERED AS { islan phy(1) action(9) initiate-remote-loopback(2) };
```

The action, force-parity-check, enables the local agent SMAP to command the initiation of a parity check test. The mode of the action is confirmed because management applications, i.e., the managing SMAP, will need to follow this action request with a GET of the parity error counter in the remote transmission peer.

```
force-parity-check   ACTION

    BEHAVIOR force-parity-check-behavior;
    MODE CONFIRMED;
    REGISTERED AS { islan phy(1) action(9) force-parity-check(3) };
```

The action, force-PHY-shutdown, enables the agent SMAP to choke the PS in the local system, if management alarms warrant such drastic action.

```
force-PHY-shutdown   ACTION

    BEHAVIOR force-PHY-shutdown-behavior;
    MODE CONFIRMED;

    REGISTERED AS { islan phy(1) action(9) force-PHY-shutdown(4) };
```

C.8.3 Notifications of the PS characteristics package

The notification, rem-loopback-failure, is sent to inform the local agent SMAP and the remote managing SMAP that a remote loopback test has failed.

```
rem-loopback-failure   NOTIFICATION

    BEHAVIOR rem-loopback-failure-behavior;

    REGISTERED AS { islan phy(1) notification(10) rem-loopback-
            failure(1) };
```

The notification, rx-tdm-parity-error-thres-exceeded, is sent to inform the local agent SMAP and the remote managing SMAP that the threshold of the rx-tdm-parity-error-counter has been exceeded.

```
rx-tdm-parity-error-thres-exceeded   NOTIFICATION

    BEHAVIOR rx-tdm-parity-error-thres-exceeded-behavior;

    REGISTERED AS { islan phy(1) notification(10) rx-tdm-parity-error-
            thres-exceeded(2) };
```

The notification, synch-loss-thres-exceeded, is sent to inform the local agent SMAP and the remote managing SMAP that the threshold of the synch-loss-counter has been exceeded.

```
synch-loss-thres-exceeded   NOTIFICATION

    BEHAVIOR synch-loss-thres-exceeded-behavior;

    REGISTERED AS { islan phy(1) notification(10) synch-loss-thres-
            exceeded(3) };
```

The notification, parity-check-fail-thres-exceeded, is sent to inform the local agent SMAP and the remote managing SMAP that the threshold of the parity-check-fail-counter has been exceeded.

```
parity-check-fail-thres-exceeded   NOTIFICATION

    BEHAVIOR parity-check-fail-thres-exceeded-behavior;

    REGISTERED AS { islan phy(1) notification(10) parity-check-fail-
            thres-exceeded(4) };
```

The notification, rem-parity-check-status, is sent to the local agent SMAP and to the remote managing SMAP to indicated the status of the parity check.

```
rem-parity-check-status   NOTIFICATION

    BEHAVIOR rem-parity-check-status-behavior;

    REGISTERED AS { islan phy(1) notification(10) rem-parity-check-
            status(5) };
```

C.9 Specifications for the PMD characteristics package

The following specifications define the information elements relevant to the pmd-characteristics package as referenced in the GDMO in 10.13.2.3 of this standard.

C.9.1 Attributes of the PMD characteristics package

The attribute, manufacturer-id, identifies the manufacturer of the implementation of the PMD characteristics package.

```
manufacturer-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-id(25) };
```

The attribute, manufacturer-product-id, identifies the product implementation model of the PMD characteristics package.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product-
            id(26) };
```

The attribute, manufacturer-product-version-id, identifies the product version of the implementation of the PMD characteristics package.

```
manufacturer-product-version-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.manufacturer-product-version-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-version-id-behavior;

    REGISTERED AS { islan phy(1) attribute(7) manufacturer-product-
            version-id(27) };
```

The attribute, loc-loopback-status, contains the status value (failure or success) of the latest local loopback requested by the local agent SMAP.

```
loc-loopback-status  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX PHY-Module.Status
    MATCHES FOR EQUALITY;
    BEHAVIOR loc-loopback-status-behavior;

    REGISTERED AS { islan phy(1) attribute(7) loc-loopback-status(28) };
```

The attribute, invalid-decode-counter, is a counter of the number of digital line code violations that have been detected in the PMD sublayer.

```
invalid-decode-counter  ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter;
    MATCHES FOR EQUALITY;
    BEHAVIOR invalid-decode-counter-behavior;

    REGISTERED AS { islan phy(1) attribute(7) invalid-decode-
            counter(29) };
```

The attribute, invalid-decode-thres, is the threshold associated with invalid-decode-counter.

```
invalid-decode-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX "ISO/IEC 10165-2: 1992": counter-Threshold;
    MATCHES FOR EQUALITY;
    BEHAVIOR invalid-decode-thres-behavior;

    REGISTERED AS { islan phy(1) attribute(7) invalid-decode-thres(30) };
```

The attribute, aggregate-trans-speed, contains the constant value expressing the aggregate speed on the point-to-point link between the ISTE and the AU.

```
aggregate-trans-speed  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX INTEGER;
    MATCHES FOR EQUALITY;
    BEHAVIOR aggregate-trans-speed;

    REGISTERED AS { islan phy(1) attribute(7) aggregate-trans-speed(31) };
```

C.9.2 Actions of the PMD characteristics package

The action, initiate-local-loopback, enables the local agent SMAP to command the initiation of a local loopback at the PS sublayer. The mode is confirmed because management applications, i.e., the managing SMAP, will need to follow this action request with a GET of the loc-loopback-status attribute.

```
initiate-local-loopback  ACTION

    BEHAVIOR initiate-local-loopback-behavior;
    MODE CONFIRMED;

    REGISTERED AS { islan phy(1) action(9) initiate-local-loopback(5) };
```

C.9.3 Notifications of the PMD characteristics package

The notification, invalid-decode-thres-exceeded, is sent to inform the local agent SMAP and the remote managing SMAP that the threshold of the invalid-decode-counter has been exceeded.

```
invalid-decode-thres-exceeded  NOTIFICATION

    BEHAVIOR parity-check-fail-thres-exceeded-behavior;

    REGISTERED AS { islan phy(1) notification(10) invalid-decode-thres-
            exceeded(6) };
```

C.10 Name binding of the ISLAN PHY entity MO

The name binding, phy-entity-name, allows the SMAP to distinguish PHY entity MOs. It also defines the containment relationship between a managed system and its PHY entity MO. It defines the create and delete operations for the PHY entity MO (but not the PHY entity itself).

```
phy-entity-name  NAME BINDING
    SUBORDINATE OBJECT CLASS islan-phy-entity;
    NAMED BY
      SUPERIOR OBJECT CLASS  "ISO/IEC 10165-2: 1992":system;
    WITH ATTRIBUTE  phy-entity-id
    BEHAVIOR phy-entity-name-behavior
    CREATE   WITH-AUTOMATIC-INSTANCE-NAMING
    DELETE   deletes-contained-objects;

    REGISTERED AS { islan phy(1) nameBinding(6) phy-entity-name(0) };
```

C.11 Associated ASN.1 specifications

The specification of the PHY packages requires the following ASN.1:

```
PHY-Module  { islan phy(1) asn1Module(2) phy-char(0) }

  DEFINITIONS IMPLICIT TAGS ::=

   BEGIN
    -- Type declarations:
    hmux-type-id         ::= INTEGER {ieee802-services-only(0),
                                      isdn-bearer-services-only(1),
                                      ieee802-and-isdn-bearer-services(2),
                                      ieee802.9-bri-and-c-channel-
                                       services(3) } ( 0...7 )
    manufacturer-id      ::= INTEGER
    manufacturer-product-
     id                  ::= INTEGER
    manufacturer-product-
     version-id          ::= INTEGER

    tx-mode-control      ::= INTEGER {ieee802-services-only(0),
                                      isdn-bearer-services-only(1),
                                      ieee802-and-isdn-bearer-services(2),
                                      ieee802.9-bri-and-c-channel-
                                       services(3) }( 0...7 )

     rx-mode-status      ::= INTEGER {ieee802-services-only(0),
                                      isdn-bearer-services-only(1),
                                      ieee802-and-isdn-bearer-services(2),
                                      ieee802.9-bri-and-c-channel-
                                       services(3) }( 0...7 )

    mode                 ::= INTEGER {ieee802-services-only(0),
                                      isdn-bearer-services-only(1),
                                      ieee802-and-isdn-bearer-services(2),
                                      ieee802.9-bri-and-c-channel-
                                       services(3) }( 0...7 )

    d-chan-rate-control ::= INTEGER { 16/64 }

    d-chan-rate-status  ::= INTEGER { 16/64 }

    hmux                ::= SEQUENCE OF
```

```
                            SEQUENCE
                             {
                                   [0] PhSAP,
                                   [1] Slot-Number,
                                   [2] Channel-Width OPTIONAL
                              }

    phsap               ::= INTEGER ( 0...255 )
    slot-number         ::= INTEGER ( 0...2499 )
    channel-width       ::= INTEGER ( 0...2499 )

    status              ::= ENUMERATED  { failure(0), success(1) }

    sync-status         ::= ENUMERATED  { failure(0), success(1) }
  END -- PHY-Module
```

C.12 Specification for the ISDN user network interface MO in the ISLAN

ISDN service provisioning requires specification of GDMO to include the LAPD entity package, the CCITT Q.93x entity package, and the user services entity package. Thus, this clause is essential to define the overall GDMO for this standard in order to facilitate full implementation of all of the services envisioned.

It should be noted that the specification of MOs for ISDN entities in 10.14 is a representative example set, which is necessary for the IEEE 802.9 ISLAN service provisioning. To understand the scope of the MO specification for the ISDN environment, see the CCITT Q.92x, Q.93x, and Q.94x Recommendations, or the equivalent ANSI documents.

The following specifications define information elements of and related to the MOs, as referenced in the GDMO in 10.14 of this standard. For consistency, the GDMOs for ISDN entities are specified in a similar manner to that of previous clauses of this annex for the MAC, HMUX, PS, and PMD sublayer MOs.

C.12.1 Attributes of the LAPD SAP MO

The LAPD address consists of both a service access point identifier (SAPI) and a technical endpoint identifier (TEI). For the purpose of the address attribute, a combined SAPI and TEI is recognized as DLCI.

```
dlce-address   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX LAPD-Attribute-Module.LAPDAddressstring;
    MATCHES FOR EQUALITY;
    BEHAVIOR dlci-address-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) dlci-address(0) };
```

C.12.2 Name binding of the LAPD SAP MO

The name binding, lapd-name, allows the LME and the SMAP to distinguish LAPD MOs. It also defines the containment relationship between an LSAP MO and the generic LAPD entity MO that contains it. It defines the creation and deletion operations for the LSAP MO (but not the LSAP itself).

```
lsap-name   NAME BINDING

    SUBORDINATE OBJECT CLASS   lsap
      NAMED BY
    SUPERIOR OBJECT CLASS generic-lapd-entity
```

```
    WITH ATTRIBUTE dlci-address;

    BEHAVIOR lsap-name-behavior;
    CREATE dlci-address;
    DELETE only if-no-contained-objects;

    REGISTERED AS { islan lapd(3) namebinding(6) lsap-name(0) };
```

The parameter, dlci-address, contains the LAPD address supplied with a create operation for an LSAP MO.

```
dlci-address  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX LSAP-attribute-Module.LAPD-Addressbitstring;

    REGISTERED AS { islan lapd(3) parameter(5) dlci-address(0) };
```

C.13 Specification of generic LAPD entity MO

The following specifications define the information elements relevant to the generic LAPD entity MO, as referenced in the GDMO in 10.14.1.1 of this standard.

C.13.1 Attributes of the generic LAPD entity MO

The attribute, lapd-entity-id, is a naming attribute for the generic LAPD entity MO.

```
lapd-entity-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module-LAPD-Entity-Id;
    MATCHES FOR Equality;
    BEHAVIOR lapd-entity-id-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) lapd-entity-id(1) };
```

The attribute, manufacturer-id, identifies the manufacturer of the implementation by the LAPD entity MO.

```
manufacturer-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.manufacturer-id;
    MATCHES FOR Equality;
    BEHAVIOR manufacturer-id-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) manufacturer-id(2) };
```

The attribute, manufacturer-product-id, identifies the product implementation model represented by the LAPD entity MO.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.manufacturer-
            product-id;
    MATCHES FOR Equality;
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) manufacturer-product-
            id(3) };
```

The attribute, manufacturer-product-version, identifies the product version of the implementation represented by the LAPD entity MO.

```
manufacturer-product-version-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.manufacturer-
            product-version-id;
    MATCHES FOR Equality;
    BEHAVIOR manufacturer-product-version-id-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) manufacturer-product-
            version-id(4) };
```

The attribute, fcs-error-count, contains a value that identifies the number of times a frame check sequence (FCS) error has occurred.

```
fcs-error-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.fcs-error-count;
    MATCHES FOR Equality;
    BEHAVIOR fcs-error-count-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) fcs_error-count(5) };
```

The attribute, invalid-frame-count, contains a value that identifies the number of times invalid frames have been received.

```
invalid-frame-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.invalid-frame-
            count;
    MATCHES FOR Equality;
    BEHAVIOR invalid-frame-count-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) invalid-frame-count(6) };
```

The attribute, max-lapd-size, contains the size of the maximum logical link protocol data unit (LPDU) that can be received.

```
max-lapd-size  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.max-lapd-size;
    MATCHES FOR Equality;
    BEHAVIOR max-lapd-size-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) max-lapd-size(7) };
```

The attribute, max-lapd-size-received, contains the size of the maximum LPDU that has been received.

```
max-lapd-size-received  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.max-lapd-size-
            received;
    MATCHES FOR Equality;
    BEHAVIOR max-lapd-size-received-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) max-lapd-size-
            received(8) };
```

The attribute, lapd-octets-received, contains the number of octets of the LAPD frame received.

```
lapd-octets-received  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.lapd-octets-
            received;
    MATCHES FOR Equality;
    BEHAVIOR lapd-octets-received-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) lapd-octets-received(9) };
```

The attribute, lapd-octets-sent, contains the number of octets of the LAPD frame sent.

```
lapd-octets-sent  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.lapd-octets-sent;
    MATCHES FOR Equality;
    BEHAVIOR lapd-octets-sent-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) lapd-octets-sent(10) };
```

The attribute, lapd-received, contains the number of LPDUs received.

```
lapd-received  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.lapd-received;
    MATCHES FOR Equality;
    BEHAVIOR lapd-received-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) lapd-received(11) };
```

The attribute, lapd-sent, contains the number of LPDUs sent.

```
lapd-sent  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.lapd-sent;
    MATCHES FOR Equality;
    BEHAVIOR lapd-sent-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) lapd-sent(12) };
```

The attribute, aged-lapd-discard-count, contains the number of LPDUs discarded from transmission.

```
aged-lapd-discard-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.aged-lapd-
            discard-count;
    MATCHES FOR Equality;
    BEHAVIOR aged-lapd-discard-count-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) aged-lapd-discard-
            count(13) };
```

The attribute, aged-lapd-discard-thres, is a counter threshold for the aged-lapd-discard-count attribute value.

```
aged-lapd-discard-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.aged-lapd-
            discard-thres;
    MATCHES FOR Equality;
    BEHAVIOR aged-lapd-discard-thres-behavior;
```

```
    REGISTERED AS { islan lapd(3) attributes(7) aged-lapd-discard-
            thres(14) };
```

The attribute, max-send-lapd-q-age, contains a value indicating the age of the LAPD in the queue.

```
max-send-lapd-q-age  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX Generic-LAPD-Attribute-Module.max-send-lapd-
            q-age;
    MATCHES FOR Equality;
    BEHAVIOR max-send-lapd-q-age-behavior;

    REGISTERED AS { islan lapd(3) attributes(7) max-send-lapd-q-age(15) };
```

C.13.2 Actions of the generic LAPD entity MO

The action, activate-lapd, activates a suspended LAPD entity.

```
activate-lapd  ACTION

    BEHAVIOR                activate-lapd-behavior;
    MODE                    CONFIRMED;
    PARAMETERS              initial-activation-values;
    WITH INFORMATION SYNTAX Parameter values;
    WITH REPLY SYNTAX       Activate response;

    REGISTERED AS { islan lapd(3) action(9) activate-lapd(0) };
```

The action, deactivate-lapd, suspends an LAPD entity.

```
deactivate-lapd  ACTION

    BEHAVIOR                deactivate-lapd-behavior;
    MODE                    NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) action(9) deactivate-lapd(1) };
```

The action, reset-lapd, resets an LAPD entity.

```
reset-lapd  ACTION

    BEHAVIOR                reset-lapd-behavior;
    MODE                    NON-CONFIRMED;
    PARAMETERS              initial-reset-values;
    WITH INFORMATION SYNTAX Parameter values;

    REGISTERED AS { islan lapd(3) action(9)-lapd(2) };
```

C.13.3 Modification of the generic LAPD entity MO

The notification, aged-lapd-discard-thres-not, notifies an agent SMAP that the threshold has been reached.

```
aged-lapd-discard-thres-not  NOTIFICATION

    BEHAVIOR aged-lapd-discard-thres-not-behavior;

    REGISTERED AS { islan lapd(3) notification(10) aged-lapd-discard-
            thres-not(0) };
```

C.13.4 Name binding entity of the generic LAPD entity MO

The name binding entity, lapd-entity-name, allows the SMAP to distinguish LAPD entity MOs. It also defines the containment relationship between its managed system and the LAPD entity MO. It defines the create and delete operations for the LAPD entity MO (but not the LAPD entity itself).

```
lapd-entity-name  NAME BINDING
    SUBORDINATE OBJECT CLASS  generic-lan-entity;
    NAMED BY SUPERIOR OBJECT CLASS-GENERIC:system;
    BEHAVIOR  lapd-entity-name-behavior;
    CREATE  lapd-entity-id
            manufacturer-id
            manufacturer-product-id
            manufacturer-product-version
            fcs-error-count
            invalid-frame-count
            max-lapd-size
            max-lapd-size-received
            lapd-octets-received
            lapd-octets-sent
            lapd-received
            lapd-sent
            aged-lapd-discard-count
            aged-lapd-discard-thres
            max-send-lapd-q-age
    DELETE  deletes-contained-objects;

    REGISTERED AS { islan lapd(3) nameBinding(6) lapd-entity-name(1) };
```

The parameter, lapd-entity-id, is the LAPD MO supplied with the create operation.

```
lapd-entity-id  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-entity-id;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-entity-id(1) };
```

The parameter, manufacturer-id, is supplied with the create operation as the manufacturer-id.

```
manufacture-id  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.manufacturer-id;

    REGISTERED AS { islan lapd(3) parameter(5) manufacturer-id(2) };
```

The parameter, manufacturer-product-id, is supplied with the create operation as the manufacturer-product-id.

```
manufacturer-product-id  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.manufacturer-product-id;

    REGISTERED AS { islan lapd(3) parameter(5) manufacturer-product-
            id(3) };
```

The parameter, manufacturer-product-version, is supplied with the create operation as the manufacturer-product-version.

```
manufacturer-product-version   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.manufacturer-product-
            version;

    REGISTERED AS { islan lapd(3) parameter(5) manufacturer-product-
            version(4) };
```

The parameter, fcs-error-count, is supplied with the create operation for specifying the number of FCS errors.

```
fcs-error-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.fcs-error-count;

    REGISTERED AS { islan lapd(3) parameter(5) fcs-error-count(5) };
```

The parameter, invalid-frame-count, is supplied with the create operation for specifying the number of invalid frames received.

```
invalid-frame-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.invalid-frame-count;

    REGISTERED AS { islan lapd(3) parameter(5) invalid-frame-count(6) };
```

The parameter, max-lapd-size, is supplied with the create operation as the maximum LAPD size.

```
max-lapd-size   PARAMETER
    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.max-lapd-size;

    REGISTERED AS { islan lapd(3) parameter(5) max-lapd-size(7) };
```

The parameter, max-lapd-size-received, is supplied with the create operation for specifying the maximum LAPD size received.

```
max-lapd-size-received   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.max-lapd-size-received;

    REGISTERED AS { islan lapd(3) parameter(5) max-lapd-size-
            received(8) };
```

The parameter, lapd-octets-received, is supplied with the create operation for specifying the number of octets received.

```
lapd-octets-received   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-octets-received;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-octets-received(9) };
```

The parameter, lapd-octets-sent, is supplied with the create operation for specifying the number of octets sent.

```
lapd-octets-sent   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-octets-sent;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-octets-sent(10) };
```

The parameter, lapd-received, is supplied with the create operation for specifying the number of LAPD frames received.

```
lapd-received   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-received;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-received(11) };
```

The parameter, lapd-sent, is supplied with the create operation for specifying the number of LAPD frames sent.

```
lapd-sent   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-sent;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-sent(12) };
```

The parameter, aged-lapd-discard-count, is supplied with the create operation for specifying the number of LAPD frames ready for discard due to aging.

```
aged-lapd-discard-count   PARAMETER
    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.aged-lapd-discard-count;

    REGISTERED AS { islan lapd(3) parameter(5) aged-lapd-discard-
            count(13) };
```

The parameter, aged-lapd-discard-thres, is supplied with the create operation and is the threshold value for the aged-lapd-discard-count attribute value.

```
aged-lapd-discard-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.aged-lapd-discard-thres;

    REGISTERED AS { islan lapd(3) parameter(5) aged-lapd-discard-
            thres(14) };
```

The parameter, max-send-lapd-q-age, is supplied with the create operation for specifying the maximum aging of a sending LAPD frame.

```
max-send-lapd-q-age   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.max-send-lapd-q-age;

    REGISTERED AS { islan lapd(3) parameter(5) max-send-lapd-q-age(15) };
```

C.14 Specification of ISLAN LAPD address package MO

The following specifications define the information elements relevant to the ISLAN LAPD address package MO, as referenced in the GDMO in 10.14.1.2 of this standard.

C.14.1 Attributes of the ISLAN address package MO

The attribute, usable-sapi-control, defines SAPI values.

```
usable-sapi-control  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.usable-sapi-
            control;
    MATCHES FOR Equality;
    BEHAVIOR usable-sapi-control-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) aged-lapd-discard-
            thres(16) };
```

The attribute, tei-assign-fail-count, counts the number of failed attempts of TEI assignments.

```
tei-assign-fail-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.tei-assign-fail-
            count;
    MATCHES FOR Equality;
    BEHAVIOR tei-assign-fail-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) tei-assign-fail-
            count(17) };
```

The attribute, tei-assign-fail-thres, is the threshold for the tei-assign-fail-count attribute value.

```
tei-assign-fail-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.tei-assign-
            fail-thres;
    MATCHES FOR Equality;
    BEHAVIOR tei-assign-fail-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) tei-assign-fail-
            thres(18) };
```

The attribute, tei-disc-request-count, specifies the number of requests for link termination, which is used for link failure identification.

```
tei-disc-request-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.tei-disc-
            request-count;
    MATCHES FOR Equality;
    BEHAVIOR tei-disc-request-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) tei-disc-request-
            count(19) };
```

The attribute, tei-disc-request-thres, is the threshold for the link termination counter, tei-disc-request-count.

```
tei-disc-request-thres  ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.tei-disc-
            request-thres;
    MATCHES FOR Equality;
    BEHAVIOR tei-disc-request-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) tei-disc-request-
            thres(20) };
```

The attribute, link-reestablishment-count, specifies the number of times that the Data Link Layer (DL) is reestablished.

```
link-reestablishment-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.link-
            reestablishment-count;
    MATCHES FOR Equality;
    BEHAVIOR link-reestablishment-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) link-reestablishment-
            count(21) };
```

The attribute, link-reestablishment-thres, is the threshold for the link-reestablishment-count attribute value.

```
link-reestablishment-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.link-
            reestablishment-thres;
    MATCHES FOR Equality;
    BEHAVIOR link-reestablishment-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) link-reestablishment-
            thres(22) };
```

The attribute, buffer-overflow-count, specifies the number of times receive frames overflow.

```
buffer-overflow-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.buffer-
            overflow-count;
    MATCHES FOR Equality;
    BEHAVIOR buffer-overflow-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) buffer-overflow-
            count(23) };
```

The attribute, buffer-overflow-thres, is the threshold for the buffer-overflow-count attribute value.

```
buffer-overflow-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.buffer-
            overflow-thres;
    MATCHES FOR Equality;
    BEHAVIOR buffer-overflow-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) buffer-overflow-
            thres(24) };
```

The attribute, lapd-retransmit-counter, specifies the number of LAPD frame retransmissions.

```
lapd-retransmit-counter  ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX  generic-LAPD-Attribute-Module.lapd-
            retransmit-counter;
    MATCHES FOR Equality;
    BEHAVIOR lapd-retransmit-counter-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) lapd-retransmit-
            counter(25) };
```

The attribute, lapd-retransmit-thres, is the threshold for lapd-retransmit-counter contents.

```
lapd-retransmit-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.lapd-retransmit-
            thres;
    MATCHES FOR Equality;
    BEHAVIOR lapd-retransmit-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) lapd-retransmit-
            thres(26) };
```

The attribute, unknown-tei-receive-count, provides the number of wrong TEIs being received.

```
unknown-tei-receive-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.unknown-tei-
            receive-count;
    MATCHES FOR Equality;
    BEHAVIOR unknown-tei-receive-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) unknown-tei-receive-
            count(27) };
```

The attribute, unknown-tei-receive-thres, is the threshold value for the unknown-tei-receive count attribute value.

```
unknown-tei-receive-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.unknown-tei-
            receive-thres;
    MATCHES FOR Equality;
    BEHAVIOR unknown-tei-receive-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) unknown-tei-receive-
            thres(28) };
```

The attribute, unknown-sapi-receive-count, provides the number of wrong SAPIs being received.

```
unknown-sapi-receive-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.unknown-sapi-
            receive-count;
    MATCHES FOR Equality;
    BEHAVIOR unknown-sapi-receive-count-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) unknown-sapi-receive-
            count(29) };
```

The attribute, unknown-sapi-receive-thres, is the threshold value for the unknown-sapi-receive count attribute value.

```
unknown-sapi-receive-thres  ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX generic-LAPD-Attribute-Module.unknown-sapi-
            receive-thres;
    MATCHES FOR Equality;
    BEHAVIOR unknown-sapi-receive-thres-behavior;

    REGISTERED AS { islan lapd(3) attribute(7) unknown-sapi-receive-
            thres(30) };
```

C.14.2 Notifications of the ISLAN address package MO

The notification, tei-assign-fail-thres-not, notifies an agent SMAP that the threshold has been reached for assignment of a TEI failure.

```
tei-assign-fail-thres-not  NOTIFICATION

    BEHAVIOR tei-assign-fail-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) tei-assign-fail-
            thres-not(1) };
```

The notification, tei-disc-request-thres-not, notifies an agent SMAP that the threshold has been reached for link termination failure.

```
tei-disc-request-thres-not  NOTIFICATION

    BEHAVIOR tei-disc-request-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) tei-disc-request-
            thres-not(2) };
```

The notification, link-reestablishment-thres-not, notifies an agent SMAP that the threshold has been reached for the number of times the link is reestablished.

```
link-reestablishment-thres-not  NOTIFICATION

    BEHAVIOR link-reestablishment-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) link-reestablishment-
            thres-not(3) };
```

The notification, buffer-overflow-thres-not, notifies an agent SMAP that the threshold has been reached in terms of the number of frame overflows.

```
buffer-overflow-thres-not  NOTIFICATION

    BEHAVIOR buffer-overflow-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) buffer-overflow-
            thres-not(4) };
```

The notification, lapd-retransmit-thres-not, notifies an agent SMAP that the threshold has been reached in terms of the number of LAPD frame retransmissions.

```
lapd-retransmit-thres-not  NOTIFICATION
```

```
    BEHAVIOR lapd-retransmit-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) lapd-retransmit-
            thres-not(5) };
```

The notification, unknown-tei-thres-not, notifies an agent SMAP that the threshold has been reached in terms of the number of wrong TEIs being received.

```
unknown-tei-thres-not   NOTIFICATION

    BEHAVIOR unknown-tei-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) unknown-tei-thres-
            not(6) };
```

The notification, unknown-sapi-thres-not, notifies an agent SMAP that the threshold has been reached in terms of the number of LAPD frames being received.

```
unknown-sapi-thres-not   NOTIFICATION

    BEHAVIOR unknown-sapi-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan lapd(3) notification(10) unknown-sapi-thres-
            not(7) };
```

C.14.3 Name binding entity of the LAPD address package MO

The name binding entity, lapd-address-name, allows the SMAP to distinguish LAPD address package MOs.

```
lapd-address-name   NAME BINDING
    SUBORDINATE OBJECT CLASS   lapd-address-package
    NAMED BY SUPERIOR OBJECT CLASS-GENERIC:system;

    BEHAVIOR lapd-entity-name-behavior;
    CREATE   usable-sapi-control
             tei-assign-fail-count
             tei-assign-fail-thres
             tei-disc-request-count
             tei-disc-request-thres
             link-reestablishment-counter
             link-reestablishment-thres
             buffer-overflow-counter
             buffer-overflow-thres
             lapd-retransmit-counter
             lapd-retransmit-thres
             unknown-tei-count
             unknown-tei-thres
             unknown-sapi-count
             unknown-sapi-thres
    DELETE   deletes-contained-objects;

    REGISTERED AS { islan lapd(3) nameBinding(6) lapd-entity-name(2) };
```

The parameter, usable-sapi-control, is the LAPD MO supplied with the create operation. This parameter provides the definition of SAPI values.

```
usable-sapi-control   PARAMETER
```

```
    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.usable-sapi-control;

    REGISTERED AS { islan lapd(3) parameter(5) usable-sapi-control(16) };
```

The parameter, tei-assign-fail-count, is the LAPD MO supplied with the create operation. This parameter provides the definition of failed TEI assignments.

```
tei-assign-fail-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.tei-assign-fail-count;

    REGISTERED AS { islan lapd(3) parameter(5) tei-assign-fail-
            count(17) };
```

The parameter, tei-assign-fail-thres, is the LAPD MO supplied with the create operation.

```
tei-assign-fail-thres  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.tei-assign-fail-thres;

    REGISTERED AS { islan lapd(3) parameter(5) tei-assign-fail-
            thres(18) };
```

The parameter, tei-disc-request-count, is the LAPD MO supplied with the create operation.

```
tei-disc-request-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.tei-disc-request-count;

    REGISTERED AS { islan lapd(3) parameter(5) tei-disc-request-
            count(19) };
```

The parameter, tei-disc-request-thres, is the LAPD MO supplied with the create operation.

```
tei-disc-request-thres  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.tei-disc-request-thres;

    REGISTERED AS { islan lapd(3) parameter(5) tei-disc-request-
            thres(20) };
```

The parameter, link-reestablishment-counter, is the LAPD MO supplied with the create operation.

```
link-reestablishment-counter  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.link-reestablishment-
            counter;

    REGISTERED AS { islan lapd(3) parameter(5) link-reestablishment-
            counter(21) };
```

The parameter, link-reestablishment-thres, is the LAPD MO supplied with the create operation.

```
link-reestablishment-thres  PARAMETER
```

```
    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.link-reestablishment-
            thres;

    REGISTERED AS { islan lapd(3) parameter(5) link-reestablishment-
            thres(22) };
```

The parameter, buffer-overflow-counter, is the LAPD MO supplied with the create operation.

```
buffer-overflow-counter   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.buffer-overflow-counter;

    REGISTERED AS { islan lapd(3) parameter(5) buffer-overflow-
            counter(23) };
```

The parameter, buffer-overflow-thres, is the LAPD MO supplied with the create operation.

```
buffer-overflow-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.buffer-overflow-thres;

    REGISTERED AS { islan lapd(3) parameter(5) buffer-overflow-
            thres(24) };
```

The parameter, lapd-retransmit-counter, is the LAPD MO supplied with the create operation.

```
lapd-retransmit-counter   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-retransmit-counter;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-retransmit-
            counter(25) };
```

The parameter, lapd-retransmit-thres, is the LAPD MO supplied with the create operation.

```
lapd-retransmit-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.lapd-thres-counter;

    REGISTERED AS { islan lapd(3) parameter(5) lapd-retransmit-
            thres(26) };
```

The parameter, unknown-tei-receive-count, is the LAPD MO supplied with the create operation.

```
unknown-tei-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.unknown-tei-count;

    REGISTERED AS { islan lapd(3) parameter(5) unknown-tei-count(27) };
```

The parameter, unknown-tei-thres, is the LAPD MO supplied with the create operation.

```
unknown-tei-thres   PARAMETER

    CONTEXT ACTION-INFO;
```

```
    WITH SYNTAX generic-LAPD-Attribute-Module.unknown-tei-thres;

    REGISTERED AS { islan lapd(3) parameter(5) unknown-tei-thres(28) };
```

The parameter, unknown-sapi-count, is the LAPD MO supplied with the create operation.

```
unknown-sapi-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.unknown-sapi-count;

    REGISTERED AS { islan lapd(3) parameter(5) unknown-sapi-count(29) };
```

The parameter, unknown-sapi-thres, is the LAPD MO supplied with the create operation.

```
unknown-sapi-thres  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX generic-LAPD-Attribute-Module.unknown-sapi-thres;

    REGISTERED AS { islan lapd(3) parameter(5) unknown-sapi-thres(30) };
```

C.15 Specification of ISDN CCITT Q.931 entity MO

The following specifications define the information elements relevant to ISLAN CCITT Q.931 protocol specific MO, as referenced in the GDMO in 10.14 of this standard.

C.15.1 Attributes of the ISDN CCITT Q.931 protocol specific MO

The attribute, signalling-id, identifies the implementation of the signalling package.

```
signalling-entity-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.signalling-
            entity-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR signalling-entity-id-behavior;

    REGISTERED AS { islan q931(4) attribute(7) signalling-entity-id(1) };
```

The attribute, manufacturer-id, identifies the manufacturer of the implementation of the signalling package.

```
manufacturer-id   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.manufacturer-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-id-behavior;

    REGISTERED AS { islan q931(4) attribute(7) manufacturer-id(2) };
```

The attribute, manufacturer-product-id, identifies the product implementation model of the signalling package.

```
manufacturer-product-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.manufacturer-
            product-id;
    MATCHES FOR EQUALITY;
```

```
    BEHAVIOR manufacturer-product-id-behavior;

    REGISTERED AS { islan q931(4) attribute(7) manufacturer-product-
            id(3) };
```

The attribute, manufacturer-product-version-id, identifies the product version of the implementation of the signalling package.

```
manufacturer-product-version-id  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.manufacturer-
            product-version-id;
    MATCHES FOR EQUALITY;
    BEHAVIOR manufacturer-product-version-id-behavior;

    REGISTERED AS { islan q931(4) attribute(7) manufacturer-product-
            version-id(4) };
```

The attribute, valid-calls-count, defines the values that represent the number of valid calls.

```
valid-calls-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.valid-calls-
            count;
    MATCHES FOR Equality;
    BEHAVIOR valid-calls-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) valid-calls-count(5) };
```

The attribute, invalid-l3-msgs-count, defines the values that represent the number of valid CCITT Q.931 messages.

```
invalid-l3-msgs-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.invalid-l3-
            msgs-count;
    MATCHES FOR Equality;
    BEHAVIOR invalid-l3-msgs-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) invalid-l3-msgs-count(6) };
```

The attribute, invalid-l3-msgs-thres, is the threshold value for the invalid-l3-msgs-count attribute value.

```
invalid-l3-msgs-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.invalid-l3-
            msgs-thres;
    MATCHES FOR Equality;
    BEHAVIOR invalid-l3-msgs-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) invalid-l3-msgs-thres(7) };
```

The attribute, all-channels-busy-count, defines the number of attempts to set up a call, but all ISDN channels are busy.

```
all-channel-busy-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.all-channel-
            busy-count;
    MATCHES FOR Equality;
    BEHAVIOR all-channel-busy-count-behavior;
```

```
    REGISTERED AS { islan q931(4) attribute(7) all-channel-busy-
            count(8) };
```

The attribute, all-channels-busy-thres, is the threshold value for the all-channels-busy-count attribute value.

```
all-channel-busy-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.all-channel-
            busy-thres;
    MATCHES FOR Equality;
    BEHAVIOR all-channel-busy-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) all-channel-busy-
            thres(9) };
```

The attribute, invalid-info-elements-count, defines the number of invalid information elements for notification of alarm.

```
invalid-info-elements-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.invalid-info-
            elements-count;
    MATCHES FOR Equality;
    BEHAVIOR invalid-info-elements-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) invalid-info-elements-
            count(10) };
```

The attribute, invalid-info-elements-thres, is the threshold value for the invalid-info-elements-count attribute value.

```
invalid-info-elements-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.invalid-info-
            elements-thres;
    MATCHES FOR Equality;
    BEHAVIOR invalid-info-elements-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) invalid-info-elements-
            thres(11) };
```

The attribute, l3-setup-received-count, defines the number of setup messages received.

```
l3-setup-received-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-setup-
            received-count;
    MATCHES FOR Equality;
    BEHAVIOR l3-setup-received-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-setup-received-
            count(12) };
```

The attribute, l3-setup-aged-count, defines a value that is the number of times the timer expiration has occurred while waiting for a setup acknowledgment for a setup message sent.

```
l3-setup-aged-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-setup-aged-
            count;
    MATCHES FOR Equality;
```

```
    BEHAVIOR l3-setup-aged-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-setup-aged-count(13) };
```

The attribute, l3-setup-aged-thres, is the threshold value for the l3-setup-aged-count attribute value.

```
l3-setup-aged-thres   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-setup-aged-
            thres;
    MATCHES FOR Equality;
    BEHAVIOR l3-setup-aged-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-setup-aged-thres(14) };
```

The attribute, l3-disconnect-aged-count, has a value that specifies the number of times connection release has not occurred while waiting for the event.

```
l3-disconnect-aged-count   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-disconnect-
            aged-count;
    MATCHES FOR Equality;
    BEHAVIOR l3-disconnect-aged-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-disconnect-aged-
            count(15) };
```

The attribute, l3-disconnect-aged-thres, is the threshold for the l3-disconnect-aged-count attribute value.

```
l3-disconnect-aged-thres   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-disconnect-
            aged-thres;
    MATCHES FOR Equality;
    BEHAVIOR l3-disconnect-aged-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-disconnect-aged-
            thres(16) };
```

The attribute, l3-release-received-counter, defines whether or not connection release has been received.

```
l3-release-received-counter   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-release-
            received-counter;
    MATCHES FOR Equality;
    BEHAVIOR l3-release-received-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-release-received(17) };
```

The attribute, l3-codeset-type, defines the current codeset in use.

```
l3-codeset-type   ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-codeset-type;
    MATCHES FOR Equality;
    BEHAVIOR l3-codeset-type-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-codeset-type(18) };
```

The attribute, l3-codeset-type-receive-error-count, defines the number of times a codeset received is in error or is an unknown codeset.

```
l3-codeset-type-receive-error-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-codeset-
            type-receive-error-count;
    MATCHES FOR Equality;
    BEHAVIOR l3-codeset-type-receive-error-count-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-codeset-type-receive-
            error-count(19) };
```

The attribute, l3-codeset-type-receive-error-thres, is the threshold value for the l3-codeset-type-receive-error-count attribute value.

```
l3-codeset-type-receive-error-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-codeset-
            type-receive-error-thres;
    MATCHES FOR Equality;
    BEHAVIOR l3-codeset-type-receive-error-thres-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-codeset-type-receive-
            error-thres(20) };
```

The attribute, l3-information-element-error, identifies the information element ID for the last detected error.

```
l3-information-element-error  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX generic-Q.931-Attribute-Module.l3-information-
            element-error;
    MATCHES FOR Equality;
    BEHAVIOR l3-information-element-error-behavior;

    REGISTERED AS { islan q931(4) attribute(7) l3-information-element-
            error(21) };
```

The attributes pertaining to CCITT Q.931 timers shall be used in accordance with the CCITT Q.93x and Q.94x Recommendations.

C.15.2 Notifications of the ISDN CCITT Q.931 MO

The notification, invalid-fail-thres-not, notifies an agent SMAP that the threshold has been reached for failing to set up an ISDN call.

```
invalid-fail-thres-not  NOTIFICATION

    BEHAVIOR invalid-fail-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) invalid-fail-thres-
            not(0) };
```

The notification, invalid-info-thres-not, notifies an agent SMAP that the threshold has been reached for number of invalid information elements.

```
invalid-info-thres-not  NOTIFICATION
```

```
    BEHAVIOR invalid-info-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) invalid-info-thres-
            not(1) };
```

The notification, l3-setup-aged-thres-not, notifies an agent SMAP that the number of waiting times for setup acknowledgment reception has reached the threshold.

```
l3-setup-aged-thres-not   NOTIFICATION

    BEHAVIOR l3-setup-aged-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) l3-setup-aged-thres-
            not(2) };
```

The notification, l3-disconnect-aged-thres-not, notifies an agent SMAP that the number of waiting times for connection release reception has reached the threshold.

```
l3-disconnect-aged-thres-not   NOTIFICATION

    BEHAVIOR l3-disconnect-aged-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) l3-disconnect-aged-
            thres-not(3) };
```

The notification, l3-codeset-type-receive-error-thres-not, notifies an agent SMAP that the number of codesets received in error has reached the threshold.

```
l3-codeset-type-received-error-thres-not   NOTIFICATION

    BEHAVIOR l3-codeset-type-received-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) l3-codeset-type-
            received-error-thres-not(4) };
```

The notification, l3-information-element-error-not, notifies an agent SMAP that an information element ID error has occurred.

```
l3-information-element-error-not   NOTIFICATION

    BEHAVIOR l3-information-element-error-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan q931(4) notification(10) l3-information-
            element-error-not(5) };
```

The notifications for MOs that have associated timers shall be specified in accordance with the CCITT Q.93x and Q.94x Recommendations.

C.15.3 Name binding entity of the ISDN CCITT Q.931 MO

The name binding entity, q931-name, allows the SMAP to distinguish ISDN CCITT Q.931 MOs.

```
q931-name  NAME BINDING
    SUBORDINATE OBJECT CLASS  q931-package
```

```
    NAMED BY SUPERIOR OBJECT CLASS-GENERIC:system;
    BEHAVIOR  q931-name-behavior;
    CREATE   signalling-entity-id
             manufacturer-id
             manufacturer-product-id
             manufacturer-product-version-id
             valid-calls-count
             invalid-l3-msgs-count
             invalid-fail-thres
             all-channels-busy-count
             all-channels-busy-thres
             invalid-info-elements-count
             invalid-info-thres
             l3-setup-received-count
             l3-setup-aged-count
             l3-setup-aged-thres
             l3-disconnect-aged-count
             l3-disconnect-aged-thres
             l3-release-received-counter
             l3-codeset-type
             l3-codeset-type-receive-error-count
             l3-codeset-type-receive-error-thres
             l3-information-element-error
    DELETE   deletes-contained-objects;
    REGISTERED AS { islan q931(4) nameBinding(6) q931-name(1) };
```

The parameter, valid-calls-count, is the CCITT Q.931 MO supplied with the create operation.

```
valid-calls-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.valid-calls-count;

    REGISTERED AS { islan q931(1) parameter(5) valid-calls-count(1) };
```

The parameter, invalid-l3-msgs-count, is the CCITT Q.931 MO supplied with the create operation.

```
invalid-l3-msgs-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.invalid-l3-msgs-count;

    REGISTERED AS { islan q931(4) parameter(5) valid-calls-count(2) };
```

The parameter, all-channels-busy-count, is the CCITT Q.931 MO supplied with the create operation.

```
all-channels-busy-count  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.all-channels-busy-count;

    REGISTERED AS { islan q931(4) parameter(5) all-channels-busy-
             count(3) };
```

The parameter, all-channels-busy-thres, is the CCITT Q.931 MO supplied with the create operation.

```
all-channels-busy-thres  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.all-channels-busy-thres;
```

```
    REGISTERED AS { islan q931(4) parameter(5) all-channels-busy-
            thres(4) };
```

The parameter, invalid-info-elements-count, is the CCITT Q.931 MO supplied with the create operation.

```
invalid-info-elements-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.invalid-info-elements-count;

    REGISTERED AS { islan q931(4) parameter(5) invalid-info-elements-
            count(5) };
```

The parameter, invalid-info-elements-thres, is the CCITT Q.931 MO supplied with the create operation.

```
invalid-info-elements-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.invalid-info-elements-thres;

    REGISTERED AS { islan q931(4) parameter(5) invalid-info-elements-
            thres(6) };
```

The parameter, l3-setup-received-count, is the CCITT Q.931 MO supplied with the create operation.

```
l3-setup-received-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-setup-received-count;

    REGISTERED AS { islan q931(4) parameter(5) l3-setup-received-
            count(7) };
```

The parameter, l3-setup-aged-count, is the CCITT Q.931 MO supplied with the create operation.

```
l3-setup-aged-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-setup-aged-count;

    REGISTERED AS { islan q931(4) parameter(5) l3-setup-aged-count(8) };
```

The parameter, l3-setup-aged-thres, is the CCITT Q.931 MO supplied with the create operation.

```
l3-setup-aged-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-setup-aged-thres;

    REGISTERED AS { islan q931(4) parameter(5) l3-setup-aged-thres(9) };
```

The parameter, l3-disconnect-aged-count, is the CCITT Q.931 MO supplied with the create operation.

```
l3-disconnect-aged-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-disconnect-aged-count;

    REGISTERED AS { islan q931(4) parameter(5) l3-disconnect-aged-
            count(10) };
```

The parameter, l3-disconnect-aged-thres, is the CCITT Q.931 MO supplied with the create operation.

```
l3-disconnect-aged-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-disconnect-aged-thres;

    REGISTERED AS { islan q931(4) parameter(5) l3-disconnect-aged-
            thres(11) };
```

The parameter, l3-release-received-counter, is the CCITT Q.931 MO supplied with the create operation.

```
l3-release-received-counter   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-release-received-counter;

    REGISTERED AS { islan q931(4) parameter(5) l3-release-received-
            counter(12) };
```

The parameter, l3-codeset-type, is the CCITT Q.931 MO supplied with the create operation.

```
l3-codeset-type   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-codeset-type;

    REGISTERED AS { islan q931(4) parameter(5) l3-codeset-type(13) };
```

The parameter, l3-codeset-type-receive-error-count, is the CCITT Q.931 MO supplied with the create operation.

```
l3-codeset-type-receive-error-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-codeset-type-receive-error-
            count;

    REGISTERED AS { islan q931(4) parameter(5) l3-codeset-type-receive-
            error-count(14) };
```

The parameter, l3-codeset-type-receive-error-thres, is the CCITT Q.931 MO supplied with the create operation.

```
l3-codeset-type-receive-error-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-codeset-type-receive-error-
            thres;

    REGISTERED AS { islan q931(4) parameter(5) l3-codeset-type-receive-
            error-thres(15) };
```

The parameter, l3-information-element-error, is the CCITT Q.931 MO supplied with the create operation.

```
l3-information-element-error   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX Q931-Attribute-Module.l3-information-element-error;
```

```
    REGISTERED AS { islan q931(4) parameter(5) l3-information-element-
            error(16) };
```

C.16 Specification of the ISDN channel MO

The following specifications define the information elements relevant to the ISDN channels within the scope of the IEEE 802.9 ISLAN interface, as referenced in the GDMO in 10.14.2.3 of this standard.

C.16.1 Attributes of the ISDN channel MO

The attribute, b1-chan-service-type, has a value that specifies the current usage of the B1 channel for circuit-mode or packet-mode services.

```
b1-chan-service-type  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.b1-chan-
            service-type;
    MATCHES FOR Equality;
    BEHAVIOR b1-chan-service-type-behavior;

    REGISTERED AS { islan chan(5) attribute(7) b1-chan-service-type(1) };
```

The attribute, b2-chan-service-type, has a value that specifies the current usage of the B2 channel for circuit-mode or packet-mode services.

```
b2-chan-service-type  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.b2-chan-
            service-type;
    MATCHES FOR Equality;
    BEHAVIOR b2-chan-service-type-behavior;

    REGISTERED AS { islan chan(5) attribute(7) b2-chan-service-type(2) };
```

The attribute, c-channels-allocated-count, has a value that specifies the current C channels allocated. This value has a range of 0–255, where 0 implies that no C channel is allocated at that instance in time.

```
c-channels-allocated-count  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.c-channels-
            allocated-count;
    MATCHES FOR Equality;
    BEHAVIOR c-channels-allocated-count-behavior;

    REGISTERED AS { islan chan(5) attribute(7) c-channels-allocated-
            count(3) };
```

The attribute, c-channels-allocated-thres, is the threshold value for the c-channels-allocated-count attribute value.

```
c-channels-allocated-thres  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.c-channels-
            allocated-thres;
    MATCHES FOR Equality;
    BEHAVIOR c-channels-allocated-thres-behavior;
```

```
    REGISTERED AS { islan chan(5) attribute(7) c-channels-allocated-
            thres(4) };
```

The attribute, max-bw-ci-channel, has a value that specifies the maximum bandwidth permitted for any given C channel.

```
max-bw-ci-channel  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.max-bw-
            ci-channel;
    MATCHES FOR Equality;
    BEHAVIOR max-bw-ci-channel-behavior;

    REGISTERED AS { islan chan(5) attribute(7) max-bw-ci-channel(5) };
```

The attribute, max-bw-ci-channel-assigned, has a value that specifies the maximum bandwidth of a currently assigned C channel.

```
max-bw-ci-channel-assigned  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.max-bw-
            ci-channel-assigned;
    MATCHES FOR Equality;
    BEHAVIOR max-bw-ci-channel-assigned-behavior;

    REGISTERED AS { islan chan(5) attribute(7) max-bw-ci-channel-
            assigned(6) };
```

The attribute, min-bw-ci-channel-assigned, has a value that specifies the minimum bandwidth of a currently assigned C channel.

```
min-bw-ci-channel-assigned  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.min-bw-
            ci-channel-assigned;
    MATCHES FOR Equality;
    BEHAVIOR min-bw-ci-channel-assigned-behavior;

    REGISTERED AS { islan chan(5) attribute(7) min-bw-ci-channel-
            assigned(7) };
```

The attribute, c-chan-service-type, has a value that specifies the service type as either circuit mode or packet mode.

```
c-chan-service-type  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.
            c-chan-service-type;
    MATCHES FOR Equality;
    BEHAVIOR c-chan-service-type-behavior;

    REGISTERED AS { islan chan(5) attribute(7) c-chan-service-type(8) };
```

The attribute, p-channel-bw, has a value that specifies current bandwidth of the P channel in terms of $n \times 64$ kb/s.

```
p-channel-bw  ATTRIBUTE

    WITH ATTRIBUTE SYNTAX GenericISDNChannelAttributeModule.p-channel-bw;
    MATCHES FOR Equality;
```

```
    BEHAVIOR p-channel-bw-behavior;

    REGISTERED AS { islan chan(5) attribute(7) p-channel-bw(9) };
```

C.16.2 Notifications of the ISDN channel MO

The notification, c-channels-allocated-thres-not, notifies an agent SMAP that the threshold has been reached and all C channels have been assigned.

```
c-channel-allocated-thres-not   NOTIFICATION

    BEHAVIOR c-channels-allocated-thres-not-behavior;
    MODE NON-CONFIRMED;

    REGISTERED AS { islan chan(5) notification(10) c-channel-allocated-
             thres-not(0) };
```

C.16.3 Name binding entity of the ISDN channel MO

The name binding entity, isdn-chan-name, allows the SMAP to distinguish ISDN channel MOs.

```
isdn-chan-name   NAME BINDING
    SUBORDINATE OBJECT CLASS isdn-channel
    NAMED BY SUPERIOR OBJECT CLASS-GENERIC:system;

    BEHAVIOR  isdn-chan-name-behavior; tab(:); 1 l
    CREATE   b1-chan-service-type
             b2-chan-service-type
             c-channels-allocated-count
             c-channels-allocated-thres
             max-bw-ci-channel
             max-bw-ci-channel-assigned
             in-bw-ci-channel-assigned
             c-channel-service-type
             p-channel-bw
             stand-rev
    DELETE   deletes-contained-objects;

    REGISTERED AS { islan chan(5) nameBinding(6) isdn-chan-name(1) };
```

The parameter, b1-chan-service-type, is the CCITT Q.931 MO supplied with the create operation.

```
b1-chan-service-type   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.b1-chan-service-type;

    REGISTERED AS { islan chan(5) parameter(5) b1-chan-service-type(1) };
```

The parameter, b2-chan-service-type, is the CCITT Q.931 MO supplied with the create operation.

```
b2-chan-service-type   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.b2-chan-service-type;

    REGISTERED AS { islan chan(5) parameter(5) b2-chan-service-type(2) };
```

The parameter, c-channels-allocated-count, is the CCITT Q.931 MO supplied with the create operation.

```
c-channels-allocated-count   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.c-channels-allocated-
             count;

    REGISTERED AS { islan chan(5) parameter(5) c-channels-allocated-
             count(3) };
```

The parameter, c-channels-allocated-thres, is the CCITT Q.931 MO supplied with the create operation.

```
c-channels-allocated-thres   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.c-channels-allocated-
             thres;

    REGISTERED AS { islan chan(5) parameter(5) c-channels-allocated-
             thres(4) };
```

The parameter, max-bw-ci-channel, is the CCITT Q.931 MO supplied with the create operation.

```
max-bw-ci-channel   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.max-bw-ci-channel;

    REGISTERED AS { islan chan(5) parameter(5) max-bw-ci-channel(5) };
```

The parameter, max-bw-ci-channel-assigned, is the CCITT Q.931 MO supplied with the create operation.

```
max-bw-ci-channel-assigned   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.max-bw-ci-channel-
             assigned;

    REGISTERED AS { islan chan(5) parameter(5) max-bw-ci-channel-
             assigned(6) };
```

The parameter, min-bw-ci-channel-assigned, is the CCITT Q.931 MO supplied with the create operation.

```
min-bw-ci-channel-assigned   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.min-bw-ci-channel-
             assigned;

    REGISTERED AS { islan chan(5) parameter(5) min-bw-ci-channel-
             assigned(7) };
```

The parameter, c-channel-service-type, is the CCITT Q.931 MO supplied with the create operation.

```
c-channel-service-type   PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.c-channel-service-type;
```

```
    REGISTERED AS { islan chan(5) parameter(5) c-channel-service-
            type(8) };
```

The parameter, p-channel-bw, is the CCITT Q.931 MO supplied with the create operation.

```
p-channel-bw  PARAMETER

    CONTEXT ACTION-INFO;
    WITH SYNTAX ISDN-Channel-Attribute-Module.p-channel-bw;

    REGISTERED AS { islan chan(5) parameter(5) p-channel-bw(9) };
```

Annex D
Recommendations for signalling procedures

(normative)

D.1 Overview

This annex contains normative information concerning signalling procedures used to prescribe bandwidth allocation between the ISTE and the access unit (AU).

This annex contains a supplemental description to augment the text of this standard for the specification of the use of CCITT Q.93x signalling procedures.

D.2 CCITT Q.931 messages

This clause provides an overview of the messages and describes the contents of each message. Each message description includes:

a) A brief description of the message direction and use
b) A table listing the information elements in the order of their appearance in the message. For each information element, the table indicates:
    1) The direction in which it may be sent; i.e., user to network (ISTE to AU), network to user (AU to ISTE), or both
    2) Whether inclusion is mandatory (M) or optional (O)

Table D.1 summarizes the messages that are used. These messages are a subset of the messages defined in CCITT Recommendation Q.931 (1993), which are necessary for call establishment and release.

Table D.1—CCITT Q.931 messages

Call establishment messages: ALERTING CALL PROCEEDING CONNECT CONNECT ACKNOWLEDGE SETUP
Call clearing messages: DISCONNECT RELEASE RELEASE COMPLETE
Miscellaneous messages: STATUS STATUS ENQUIRY

D.2.1 Alerting

This message is sent by the called user to the network and by the network to the calling user to indicate that called user alerting has been initiated.

Message type: ALERTING
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
(Extended) channel identification	Both	O (see Note 1)
User-to-user	Both	O (see Note 2)

NOTES
1— This is mandatory if this message is the first message in response to SETUP. If a C channel call is in progress, then the extended channel identification information element is used preceded by a shift to Codeset 7; otherwise, if it is either a B1 or B2 channel call, then the Codeset 0 channel identification information element is used.
2— Included to support user-to-user signalling.

D.2.2 Call proceeding

This message is sent by the called user to the network or by the network to the calling user to indicate that requested call establishment has been initiated.

Message type: CALL PROCEEDING
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
(Extended) channel identification	Both	O (see Note)

NOTE— This is mandatory if this message is the first message in response to SETUP. If a C channel call is in progress, then the extended channel identification information element is used preceded by a shift to Codeset 7; otherwise, if it is either a B1 or B2 channel call, then the Codeset 0 channel identification information element is used.

D.2.3 Connect

This message is sent by the called user to the network and by the network to the calling user to indicate call acceptance by the called user.

Message type: CONNECT
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
(Extended) channel identification	Both	O (see Note 1)
Low-layer compatibility	Both	O (see Note 2)
User-to-user	Both	O (see Note 3)

NOTES
1— This is mandatory if this message is the first message in response to SETUP. If a C channel call is in progress, then the extended channel identification information element is used preceded by a shift to Codeset 7; otherwise, if it is either a B1 or B2 channel call, then the Codeset 0 channel identification information element is used.
2 —Included to provide low-layer compatibility information to the end station.
3—Included to support user-to-user signalling.

D.2.4 Connect acknowledge

This message is sent by the network to the called user to indicate that the user has been awarded the call. It may also be sent by the calling user to the network to allow symmetrical call control procedures.

Message type: CONNECT ACKNOWLEDGE
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M

D.2.5 Disconnect

This message is sent by the user to request the network to clear an end-to-end connection or is sent by the network to indicate that the end-to-end connection has been cleared.

Message type: DISCONNECT
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
Cause	Both	M
User-to-user	Both	O (see Note)
NOTE—Included to support user-to-user signalling.		

D.2.6 Release

This message is sent by the user or the network to indicate that the equipment sending the message has disconnected the channel and intends to release the channel and the call reference, and that the receiving equipment should release the channel and prepare to release the call reference after sending RELEASE COMPLETE.

Message type: RELEASE
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
Cause	Both	O (see Note 1)
User-to-user	Both	O (see Note 2)
NOTES 1—This is mandatory in the first call clearing message, including when the RELEASE message is sent as a result of an error handling condition. 2—Included to support user-to-user signalling.		

D.2.7 Release complete

This message is sent by the user or the network to indicate that the equipment sending the message has released the channel and the call reference, the channel is available for reuse, and the receiving equipment should release the call reference.

Message type: RELEASE COMPLETE
Direction: both

Information element	Direction	Type
Protocol discriminators	Both	M
Call reference	Both	M
Message type	Both	M
Cause	Both	O (see Note 1)
User-to-user	Both	O (see Note 2)
NOTES 1—This is mandatory in the first call clearing message, including when the RELEASE COMPLETE message is sent as a result of an error handling condition. 2—Included to support user-to-user signalling.		

D.2.8 Setup

This message is sent by the calling user to the network and the network to the called user to initiate call establishment.

Message type: SETUP
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M
(Extended) bearer capability	Both	M (see Note 1)
(Extended) channel identification	Both	O (see Note 2)
Calling party number	Both	O (see Note 3)
Calling party subaddress	Both	O (see Note 4)
Called party number	Both	M
Called party subaddress	Both	O (see Note 5)
Low-layer compatibility	Both	O (see Note 6)
High-layer compatibility	Both	O (see Note 7)
User-to-user	Both	O (see Note 8)

NOTES
1—If a C channel call is under progress, then the extended bearer capability information element is used preceded by a shift to Codeset 7; otherwise, if it is either a B1 or B2 channel call, then the Codeset 0 bearer capability information element is used.
2—This is mandatory in the network-to-user direction. It is included in the user-to-network direction if the user wants to indicate the channels. If it is not included, its absence is interpreted as "any channel acceptable." If a C channel call is in progress, then the extended channel identification information element is used preceded by a shift to Codeset 6; otherwise, if it is either a B1 or B2 channel call, then the Codeset 0 channel identification information element is used.
3—This may be included by the calling user or the network to identify the calling user.
4—This is included in the user-to-network direction when the calling user wants to indicate the calling party subaddress. It is included in the network-to-user direction if the calling user included a calling party subaddress information element in the SETUP message.
5—This is included in the user-to-network direction when the calling user wants to indicate the called party subaddress. It is included in the network-to-user direction if the calling user included a called party subaddress information element in the SETUP message.
6—Included to provide low-layer compatibility information to the end station.
7—Included to provide high-layer compatibility information to the end station.
8—Included to support user-to-user signalling.

D.2.9 Status

This message is sent by the user or the network at any time during a call to report certain error conditions.

Message type: STATUS
Direction: both

Information element	Direction	Type
Protocol discriminators	Both	M
Call reference	Both	M
Message type	Both	M
Cause	Both	M
Call state	Both	M

D.2.10 Status enquiry

This message is sent by the user or the network at any time to solicit a STATUS message from the peer Layer 3 entity. Sending a STATUS message in response to a STATUS ENQUIRY message is mandatory.

Message type: STATUS ENQUIRY
Direction: both

Information element	Direction	Type
Protocol discriminator	Both	M
Call reference	Both	M
Message type	Both	M

D.3 Information elements

This clause provides a description of the new information elements introduced in Chapter 11.

D.3.1 Extended bearer capability

The extended bearer capability information element completely describes the bearer service to be provided by the network. Bandwidth negotiation is also possible by specifying a maximum and minimum transfer rate. The coding of this information element is shown in table D.2.

Table D.2—Extended bearer capability information element (IE)

<table>
<tr><td>8</td><td colspan="6"></td><td>1</td><td>Octets</td></tr>
<tr><td>0</td><td colspan="7">Extended bearer capability IE
0 0 0 0 1 0 1</td><td>1</td></tr>
<tr><td colspan="8">Length of the extended bearer capability contents</td><td>2</td></tr>
<tr><td>1 ext.</td><td colspan="2">Coding standard</td><td colspan="5">Information transfer capability</td><td>3</td></tr>
<tr><td>0 ext.</td><td colspan="2">Transfer mode</td><td colspan="3">Reserved
0 0 0</td><td colspan="2">Symmetry</td><td>4</td></tr>
<tr><td>1 ext.</td><td colspan="2">Structure</td><td colspan="3">Configuration</td><td colspan="2">Establishment</td><td>4.1</td></tr>
<tr><td>1 ext.</td><td>0 Res</td><td></td><td colspan="5">Information transfer base rate</td><td>5</td></tr>
<tr><td>0 ext.</td><td colspan="7">Maximum transfer rate multiplier
(Origination to Destination)</td><td>6</td></tr>
<tr><td>0/1 ext.</td><td colspan="7">Maximum transfer rate multiplier (contd.) (MSB)
(Origination to Destination)</td><td>6.1</td></tr>
<tr><td>0/1 ext.</td><td colspan="7">Minimum transfer rate multiplier
(Origination to Destination)</td><td>6.2
(see Note 1)</td></tr>
<tr><td>1 ext.</td><td colspan="7">Minimum transfer rate multiplier (contd.) (MSB)
(Origination to Destination)</td><td>6.3
(see Note 1)</td></tr>
<tr><td>0 ext.</td><td colspan="7">Maximum transfer rate multiplier
(Destination to Origination)</td><td>7
(see Note 2)</td></tr>
<tr><td>0/1 ext.</td><td colspan="7">Maximum transfer rate multiplier (contd.) (MSB)
(Destination to Origination)</td><td>7.1
(see Note 2)</td></tr>
<tr><td>0/1 ext.</td><td colspan="7">Minimum transfer rate multiplier
(Destination to Origination)</td><td>7.2
(see Notes 1 & 2)</td></tr>
<tr><td>1 ext.</td><td colspan="7">Minimum transfer rate multiplier (contd.) (MSB)
(Destination to Origination)</td><td>7.3
(see Notes 1 & 2)</td></tr>
<tr><td colspan="8">NOTES
1—These fields shall be present only in the user-to-network direction and are used for bandwidth negotiation.
2—These fields are present only if the symmetry is bidirectional asymmetric. The use of asymmetric bandwidth support is a matter for further study.</td><td></td></tr>
</table>

The encoding for the extended bearer capability IE is described below:

Coding standard (octet 3): Bits 7 6 ------ 0 0 CCITT standardized coding as described below All other values are reserved.
Information transfer capability (octet 3): Bits 5 4 3 2 1 --------------- 0 1 0 0 0 unrestricted digital information All other values are reserved.
Transfer Mode (octet 4): Bits 7 6 ------ 0 0 circuit mode 1 1 IEEE 802.9 packet mode (Used to support CCITT Q.931 procedures for P channel signalling.) All other values are reserved.
Symmetry (octet 4): Bits 2 1 ------ 0 0 bidirectional symmetric 1 0 bidirectional asymmetric 0 1 unidirectional forward (origination to destination) 1 1 unidirectional backward (destination to origination)
Structure (octet 4.1): Bits 7 6 5 --------- 0 0 1 8 kHz integrity 1 0 0 service data unit integrity (for IEEE 802.9 packet mode transfer) All other values are reserved.
Configuration (octet 4.1): Bits 4 3 ------ 0 0 point-to-point All other values are reserved.
Establishment (octet 4.1): Bits 2 1 ------ 0 0 demand All other values are reserved.

Information transfer base rate (octet 5): Bits 6 5 4 3 2 1 ------------------ 0 0 0 0 0 0   64 kb/s 0 0 0 0 0 1   128 kb/s 0 0 0 0 1 0   384 kb/s 0 0 0 0 1 1   1536 kb/s 0 0 0 1 0 0   1920 kb/s 0 0 0 1 0 1   6144 kb/s All other values are reserved.
Maximum/minimum transfer rate multiplier (octets 6–7): This field describes the binary encoding of the maximum/minimum transfer rate multiplier value, n, for n times the information transfer base rate. These fields are used for bandwidth negotiation between the user (ISTE) and the network (AU). The maximum transfer rate multiplier combined with the information transfer base rate, is the preferred bandwidth requested by the caller. The minimum transfer rate multiplier combined with the information transfer base rate is the lowest possible bandwidth that the caller will accept (this field is present only in the user-to-network direction). Depending on the resource availability, the network (AU) end will allocate bandwidth between these two limits, with a value that is closest to the maximum requested, and is not lower than the minimum. If no value is specified for the minimum transfer rate multiplier, it defaults to the maximum transfer rate value. These values can be specified in both directions for supporting bidirectional asymmetric bandwidth. The most significant byte (MSB) of the multiplier value is specified in the higher numbered octet.

D.3.2 Extended channel identification

The extended channel identification information element is used to identify the channels that are controlled by these signalling procedures. The coding of this IE is shown in table D.3.

Table D.4 describes the channel number assignment. Table D.5 describes the bit positions in the slot map corresponding to the time slots (of specified channel granularity) occupied by a call, which are set to '1.' All other bits are set to '0.'

D.3.3 Cause

The diagnostics field for cause numbers 57, 58, and 65 are described below. These three cause values relate to the extended bearer capability IE. Specifically, two new attribute numbers, 10 and 11, which pertain to the extended information transfer rates in both directions (origination to destination and destination to origination), have been introduced. These would replace attribute numbers 3 and 8, which are used to represent the narrowband information transfer rates. The encoding of the diagnostics field is shown in table D.6.

In the case of the extended bearer capability IE, both the rejected attribute and the available attribute fields are encoded as shown in table D.7.

A new cause number, 89, is introduced for identifying the channels associated with the extended channel identification IE as shown in table D.8. This replaces cause number 82, which is used with the existing Codeset 0 channel identification IE. The diagnostics field for cause number 89 requires that the channel be

Table D.3—Extended channel identification IE

<table>
<tr><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>Octets</td></tr>
<tr><td>0</td><td colspan="7">Extended channel identification
0 0 1 1 0 0 1</td><td>1</td></tr>
<tr><td colspan="8">Length of the extended bearer capability contents</td><td>2</td></tr>
<tr><td>1 ext</td><td>Interface ID present</td><td>Interface type</td><td>0 Reserved</td><td>Preferred/ Exclusive</td><td>D Channel indicator</td><td colspan="2">Information Channel selection</td><td>3</td></tr>
<tr><td>1 ext.</td><td colspan="2">Coding standard</td><td colspan="5">Reserved
0 0 0 0 0</td><td>4</td></tr>
<tr><td>1 ext.</td><td colspan="2">Number/map</td><td colspan="5">Channel granularity</td><td>5</td></tr>
<tr><td colspan="8">Length of channel number/slot map field
(Origination to Destination)</td><td>6</td></tr>
<tr><td colspan="8">Channel number/slot map field
(Origination to Destination)</td><td>7</td></tr>
<tr><td colspan="8">Length of channel number/slot map field
(Destination to Origination)</td><td>(see Notes 1 and 2)</td></tr>
<tr><td colspan="8">Channel number/slot map
(Destination to Origination)</td><td>(see Notes 1 and 2)</td></tr>
<tr><td colspan="8">NOTES
1—These fields are present only if the symmetry is bidirectional asymmetric. The use of asymmetric bandwidth support is a matter for further study.
2—The channel number/length pairs can be repeated to accommodate noncontiguous channel allocation.</td><td></td></tr>
</table>

identified. This can be done by using the same format that has been used in the extended channel identification field, as shown in table D.9.

D.3.4 Low layer compatibility

The low layer compatibility IE is transmitted by the CCITT Q.931-based call control entity during call setup. To support the IEEE 802.9 C channel call setup requirements, the encoding for the low layer compatibility IE is shown in table D.10. User-specified Layer 2 and Layer 3 protocol information can also be exchanged between the communicating ISTEs.

The purpose of the low layer compatibility IE is to provides a means to be used for compatibility checking by the addressed entity. The low layer compatibility IE is transferred transparently between the call originating entity and the addressed entity.

The coding of this information element is shown in table D.10.

Interface identifier present (octet 3): Bit 7 ----- 0 interface implicitly identified
Interface type (octet 3): Bit 6 ----- 1 other interface (non-BRI)
Preferred/exclusive (octet 3): Bit 4 ----- 0 indicated channels are preferred 1 exclusive: only the indicated channels are acceptable
D channel indicator (octet 3): Bit 3 ----- 0 the channel identified is not the D channel
Information channel selection (octet 3): Bits 2 1 ------ 0 0 no channels indicated 0 1 as indicated in the following octets
Coding standard (octet 4): Bits 7 6 ------ 0 0 CCITT standardized coding
Channel number/map (octet 5): Bit 7 --- 0 channels are indicated by a number/length pair in the following octets 1 channels are indicated by a slot map in the following octets
Channel granularity (octet 5): Bits 6 5 4 3 2 1 ------------------ 0 0 0 0 0 0 64 kb/s 0 0 0 0 0 1 128 kb/s 0 0 0 0 1 0 384 kb/s 0 0 0 0 1 1 1536 kb/s 0 0 0 1 0 0 1920 kb/s 0 0 0 1 0 1 6144 kb/s All other values are reserved.
Channel number: This field describes the binary number assigned to the starting time slot(s) of the specified channel granularity. The length field specifies the contiguous time slots (of the specified granularity) from the starting time slot, which constitute the required bandwidth. The channel number/length pairs can be repeated to accommodate noncontiguous channel allocation.

Table D.4—Channel number assignment

8 1	Octets
Channel number	7.1
Channel number (contd.) (MSB)	7.2 (see Note)
Length	7.3
NOTE—This octet (7.2) shall be encoded as all zeros.	

Table D.5—Slot map for 4.096 Mb/s interface

8							1	Octets
64	63	62	61	60	59	58	57	7.1
56	55	54	53	52	51	50	49	7.2
48	47	46	45	44	43	42	41	7.3
40	39	38	37	36	35	34	33	7.4
32	31	30	29	28	27	26	25	7.5
24	23	22	21	20	19	18	17	7.6
16	15	14	13	12	11	10	9	7.7
8	7	6	5	4	3	2	1	7.8

NOTES
1—This slot map table illustrates a 4096 kb/s interface at 64 kb/s granularity.
2—This slot map would be extended for the case of the 20.48 Mb/s interface.

Table D.6—Encoding of diagnostics field for cause IE

8	1	Octets
0/1 ext.	Attribute number	5
0/1 ext.	Rejected attribute	5a
1 ext.	Available attribute	5b

Attribute number (octet 5):
Bits
7 6 5 4 3 2 1 No.
------------------- ----
0 1 1 1 0 1 0 10 Extended information transfer rate (origination to destination)
0 1 1 1 0 1 1 11 Extended information transfer rate (destination to origination)

Table D.7—Encoding of the rejected attribute and available attribute fields for cause IE associated with extended bearer capability exchange

8		1	Octets
0 ext.	0 Reserved	Information transfer base rate	5
0 ext.	Rejected attribute		5a
0/1 ext.	Available attribute		5b

Table D.8—Cause value associated with extended channel identification IE

Cause class	Cause value	Cause no.	Cause	Diagnostics
765 101	4321 1001	89	Identified extended channel does not exist	Channel identity

Table D.9—Encoding of channel attribute field for cause IE associated with extended channel identification exchange

8		1	Octets
0 Reserved	Number/map	Channel granularity	5a/5b
Length of channel number/slot map field			5a/5b
Channel number/slot map			5a/5b

D.4 Circuit-switched calls

This clause describes the CCITT Q.931 call control states for circuit-switched B and C channel calls between the AU and the ISTE. These states may be effected by a variety of topologies, whereas the ISTE may be a terminal equipment adapter attached to a variety of equipment or it may be an integrated intelligent communication adapter within the workstation. Nevertheless, the states and signals are explained as the message flow between an ISTE application and an AU.

Table D.10—Encoding of the low-layer compatibility IE

8		1	Octets
0	Low layer compatibility IE 1 1 1 1 0 0 0		1
Length of the low layer compatibility contents			2
1 ext	Coding standard	Information transfer capability	3
1 ext.	Transfer mode	Reserved 1 1 1 1 1	4
1 ext.	Layer 1 ident. 0 1	User information Layer 1 protocol	5
0/1 ext.	Layer 2 ident. 1 0	User information Layer 2 protocol	6
1 ext.	User-specified Layer 2 protocol information		6a
0/1 ext.	Layer 3 ident. 1 1	User-specified Layer 3 protocol information	7
1 ext.	User-specified Layer 3 protocol information		7a

D.4.1 Call states at the user side of the interface

The call states that may exist on the ISTE application side of the ISTE-to-AU interface are defined below.

D.4.1.1 Null state

No call exists between the ISTE terminal adapter (TA) and the AU.

D.4.1.2 Call initiated state (U1)

This state exists for an outgoing call when the ISTE application requests call establishment from the AU (network or IEEE 802.9 subnetwork).

D.4.1.3 Outgoing call proceeding state (U3)

This state exists for an outgoing call when the ISTE application has received acknowledgment that the AU has received all call information necessary to effect call establishment.

D.4.1.4 Call delivered state (U4)

This state exists for an outgoing call when the calling ISTE application has received an indication that the remote (ISTE or network) user alerting has been initiated.

Coding standard (octet 3): Bits 7 6 ------ 0 0 CCITT standardized coding All other values are reserved.
Information transfer capability (octet 3): Bits 5 4 3 2 1 --------- 0 1 0 0 0 unrestricted digital information All other values are reserved.
Transfer mode (octet 4): Bits 7 6 ------ 0 0 circuit mode 1 1 IEEE 802.9 packet mode (Used to support CCITT Q.931 procedures for P channel signalling) All other values are reserved.
User information Layer 1 protocol (octet 5): As specified in CCITT Recommendation Q.931.
User information Layer 2 protocol (octet 6): In addition to the ones specified in CCITT Recommendation Q.931, the following codepoint is introduced. Bits 5 4 3 2 1 ---------- 1 0 0 0 0 User specified (When this coding is specified, octet 6a will include user-specified Layer 2 information.)
User information Layer 2 protocol (octet 6a): The coding of this field depends on user-defined requirements.
User information Layer 3 protocol (octet 7): In addition to the ones specified in CCITT Recommendation Q.931, the following codepoint is introduced. Bits 5 4 3 2 1 ---------- 1 0 0 0 0 User specified (When this coding is specified, octet 7a will include user-specified Layer 3 information.)
User information Layer 3 protocol information (octet 7a): The coding of this field depends on user-defined requirements.

D.4.1.5 Call present (U6)

This state exists for an incoming call from the AU when the ISTE application has received a call establishment request but has not yet responded.

D.4.1.6 Call received state (U7)

This state exists for an incoming call when the ISTE application has indicated alerting but has not yet answered.

D.4.1.7 Connect request state (U8)

This state exists for an incoming call when the ISTE application has answered the call and is waiting to be awarded the call by the AU.

D.4.1.8 Incoming call proceeding state (U9)

This state exists for an incoming call to the ISTE application when the ISTE application has sent an acknowledgment that the ISTE application has received all call information necessary to effect call establishment.

D.4.1.9 Active state (U10)

This state exists for an incoming call when the ISTE application has received an acknowledgment from the AU that the ISTE application has been awarded the call. This state exists for an outgoing call when the ISTE application has received an indication that the remote ISTE application has answered the call.

D.4.1.10 Disconnect request state (U11)

This state exists when the ISTE application has requested the AU to clear the end-to-end connection (if any) and is waiting for the response for disconnect indication (U12).

D.4.1.11 Disconnect indication state (U12)

This state exists when the ISTE application has received an invitation to disconnect because the AU has disconnected the end-to-end connection (if any).

D.4.1.12 Release request (U19)

This state exists when the ISTE application has requested the network to release and is waiting for a response.

D.4.2 Call states at the AU side of the interface

D.4.2.1 Null state (N0)

No call exists at this time.

D.4.2.2 Call initiated state (N1)

This state exists for an outgoing call when the AU has received a call establishment request but has not yet responded.

D.4.2.3 Outgoing call proceeding state (N3)

This state exists for an outgoing call when the AU has sent an acknowledgment that the AU has received all call information necessary to effect call establishment.

D.4.2.4 Call delivered state (N4)

This state exists for an outgoing call when the AU has indicated that remote ISTE user alerting has been initiated.

D.4.2.5 Call present state (N6)

This state exists for an incoming call when the AU has sent a call establishment request but has not yet received a satisfactory response.

D.4.2.6 Call received state (N7)

This state exists for an incoming call when the AU has received an indication that the user is alerting but has not yet received an answer.

D.4.2.7 Connect request state (N8)

This state exists for an incoming call when the AU has received an answer but the AU has not yet awarded the call.

D.4.2.8 Incoming call proceeding state (N9)

This state exists for an incoming call when the AU has received an acknowledgment that the ISTE application has received all call information necessary to effect call establishment.

D.4.2.9 Active state (N10)

This state exists for an incoming call when the AU has awarded the call to the called ISTE application. This state exists for an outgoing call when the AU has indicated that the remote ISTE application has answered the call.

D.4.2.10 Disconnect request state (N11)

This state exists when the network has received a request from the ISTE application to clear the end-to-end connection (if any).

D.4.2.11 Disconnect indication state (N12)

This state exists when the AU has disconnected the end-to-end connection (if any) and has sent an invitation to disconnect the ISTE application–to-AU connection.

D.4.2.12 Release request state (N19)

This state exists when the AU has requested the ISTE application to release and is waiting for a response.

D.5 Supplementary services

Supplementary services are services that are extensions to the basic call control abilities of CCITT Q.931 that provide for special types of communication services or for special types of applications communication requirements. Depending on the abilities of the local network provider, supplementary services may extend end-to-end through the wide area network (WAN).

In most cases, it is envisioned that supplementary services would be used for localized special client server applications between endpoints and the AU. The IEEE 802.9 network would be used to implement vendor-specialized networks supporting services in manufacturing, computer aided design, video conferencing hubs, etc. These services would extend the abilities to coordinate information retrieval and delivery from both WAN and LAN sources and integrate the information flow to the ISTE.

Some basic examples of supplementary services include, but are not limited to:

— Audio conference calling
— Multimedia conferencing
— Requesting AU for synchronization among multiple active channels
— Call hold
— Time-dependent call activation—Automated voice message deliveries
— List calling

D.5.1 Audio conference calling

In this supplementary call service, a single call request would be sent to the AU for multiple destinations required for a single service. The calls could be either internal IEEE 802.9 network nodes only, or to internal and external nodes.

The IS workstation would send a CCITT Q.931 supplementary call request message to the IEEE 802.9 AU. The call request would be received by the AU and then the AU would issue multiple call requests to each of the requested parties bearing the same calling station identifier and terminal equipment identifier (TEI). The combination of the station identifier and TEI indicates to the AU which endpoint and TE channel are requesting the conferencing service. Once all call completion messages have been received by the AU, the AU would then connect the calls to the conference calling station.

D.5.2 Multimedia conferencing

This application introduces the signalling requirement that there be coordination between the transmission needs of the C channel(s) and a B channel, and a connectionless P channel assignment. Multimedia supplementary services are defined as one or more channels bearing services for video transmission, one or more B channels carrying the audio portion of the communication, and possibly one or two B or C channels carrying the imaging data during the conference call. These channels may originate from different sources but the delivery of information is coordinated or synchronized by the AU to the IS workstation. The information delivery would then appear to originate from a single source.

These are only examples of supplementary services that could be incorporated in an IEEE 802.9 network. All integrated network stations would still comply with the basic IEEE 802.9 specification, with extensions to call control.

Annex E
Architecture of the access unit (AU) and guidelines for implementation

(informative)

This annex provides reference models of common configurations. The reference configuration diagrams in this annex are to be used as guidelines for implementation.

E.1 Bearer service in a Mode 0 environment

Figure E.1 illustrates how a packet bearer service is supported across a subnetwork consisting of the integrated services terminal equipment (ISTE) and an access unit (AU) for communication with another IEEE 802.x local area network (LAN). Within the hybrid multiplexer control (HMC) field of the IEEE 802.9 time division multiplexer (TDM) frame, there is the ability to specify the configuration under which local bandwidth management is to be effected between the ISTE and the AU within the IEEE 802.9 management and signalling procedures. The configuration types are defined as "modes" of operation. Figure E.1 depicts a topology with Mode 0 operation. In this mode, the TE can only use the "IEEE 802 LAN" component of the bandwidth. Thus, only the P channel portion of the bandwidth is actively used. This means that:

a) The ISTE does not use the B1 or B2 fixed channel slots.
b) The ISTE does not negotiate for the payload to be configured into C channels.
c) The payload is totally devoted to an IEEE 802 LAN communication over the P channel.

With reference to figure E.1, when an ISTE application entity wishes to transmit information, it presents it to the IEEE 802.9 MAC. The MAC sublayer in turn presents the packetized information flow to the P channel. The ISTE's Physical (PHY) Layer exchanges this information with the AU's PHY Layer. Upon reception, the AU's MAC sublayer examines the MAC address on the received frame to determine whether the information packet is destined for another ISTE connected to this local AU or if it is destined for connection to another IEEE 802.x LAN. If the data is destined for another LAN, the packets are sent over an IEEE 802.1D-compliant bridge circuit.

The main intent of Mode 0 is to enable the implementor the ability to offer lower cost, lower functionality products that solely support a traditional IEEE 802 data LAN application.

E.2 Bearer service in a Mode 1 environment

Figure E.2 illustrates how an integrated services digital network (ISDN) basic rate circuit-switched and packet-switched bearer service is supported across a subnetwork consisting of the ISTE and an AU for communication with the ISDN. Within the HMC field of the IEEE 802.9 TDM frame, there is the ability to specify the configuration under which local bandwidth management is be effected between the ISTE and the AU within the IEEE 802.9 management and signalling procedures. The configuration types are defined as "modes" of operation. Figure E.2 depicts a topology with Mode 1 operation. In this mode, the TE can only use the "ISDN basic rate" component of the bandwidth. Thus, only the fixed D channel, B1 channel, and B2 channel portions of the bandwidth are actively used. This means that:

a) The ISTE may use the B1 or B2 fixed channel slots.
b) The ISTE does not use the P channel.
c) The ISTE does not negotiate for the payload to be configured into the C channels.
d) The need for support of the ISDN CCITT Q.93x based signalling is essential for the negotiation of call setup agreements for the establishment of B1 and B2 bearer channel usage

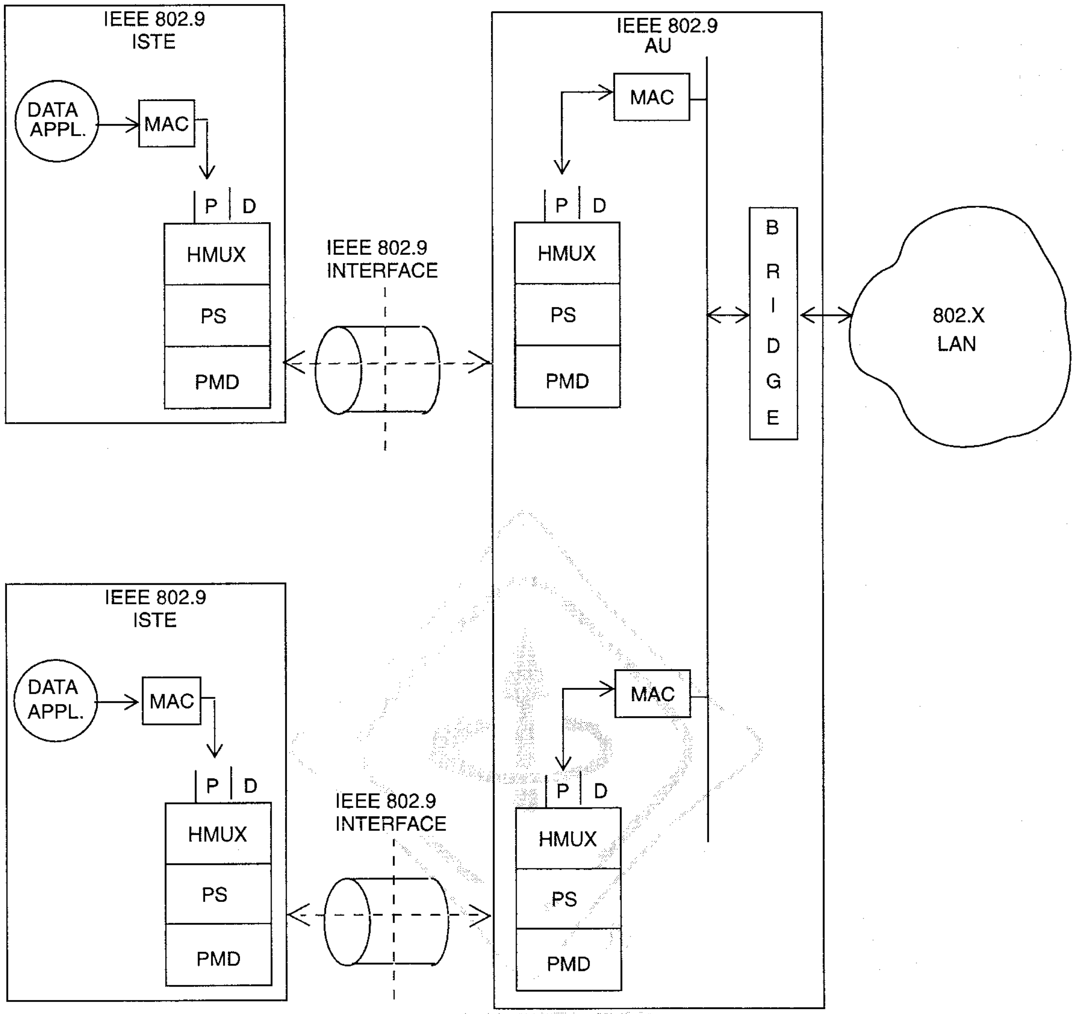


Figure E.1—Bearer service in a Mode 0 environment

With reference to figure E.2, whenever an ISTE application entity wishes to transmit information, it has its call control module communicate via CCITT Q.93x messages over the D channel to the ISDN. Once the network has granted a "connection," the ISTE's application entity applies circuit-switched data/voice/video to the respective B channel. The application entity applies packet-switched communication to either the B or the D channel as negotiated via CCITT Q.93x.

In the AU, the involvement of the connection management can be via a simple multiplexer arrangement wherein the D channel messages are simply routed on toward the ISDN. Alternatively, the AU may be part of the ISDN switching network where call requests are processed. In this second situation, the AU can provide local circuit- and packet-switched connections from one ISTE port to another ISTE port.

The intent of Mode 1 is to enable an implementor the ability to offer lower cost, lower functionality products that solely support a traditional ISDN application.

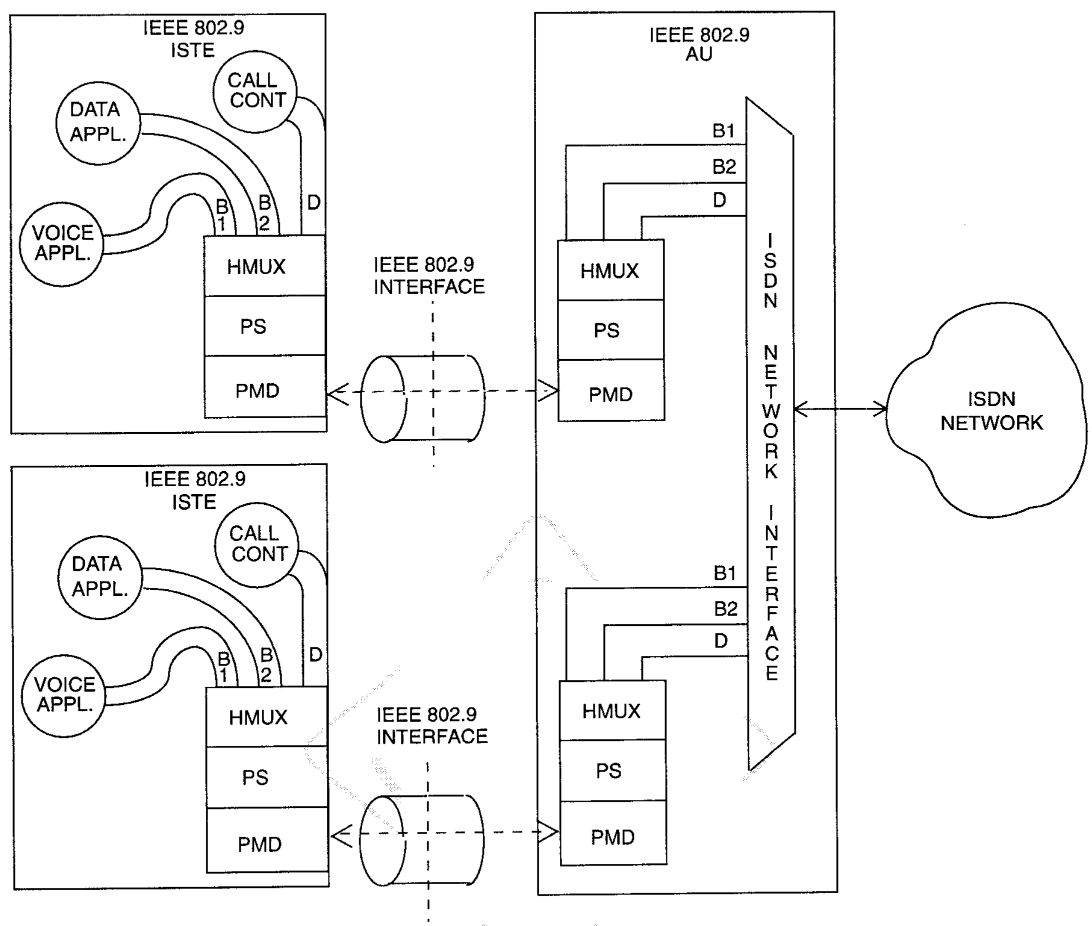


Figure E.2—Bearer service in a Mode 1 environment

E.3 Bearer service in a Mode 2 environment

Figure E.3 illustrates an environment wherein the complete set of ISDN basic rate and IEEE 802 LAN services may be simultaneously supported. Within the HMC field of the IEEE 802.9 TDM frame, there is the ability to specify the configuration under which local bandwidth management is be effected between the ISTE and the AU within the IEEE 802.9 management and signalling procedures. The configuration types are defined as "modes" of operation. Figure E.3 depicts a topology with Mode 2 operation. The local call control entity shall use CCITT Q.93x to negotiate the B channel services with the AU. This means that:

a) The ISTE may use the B1 or B2 fixed channel slots.
b) The ISTE does not negotiate for the payload to be configured into C channels.
c) The payload is totally devoted to a IEEE 802 LAN communication over the P channel.
d) The need for support of the ISDN CCITT Q.93x based signalling is essential for the negotiation of call setup agreements for the establishment of B1 and B2 bearer channel usages.

In the AU, the involvement of the connection management can be via a simple multiplexer arrangement wherein the D channel messages are simply routed on toward the ISDN. Alternatively, the AU may be part of the ISDN switching network where call requests are processed. In this second situation, the AU can provide local circuit- and packet-switched connections from one ISTE port to another ISTE port.

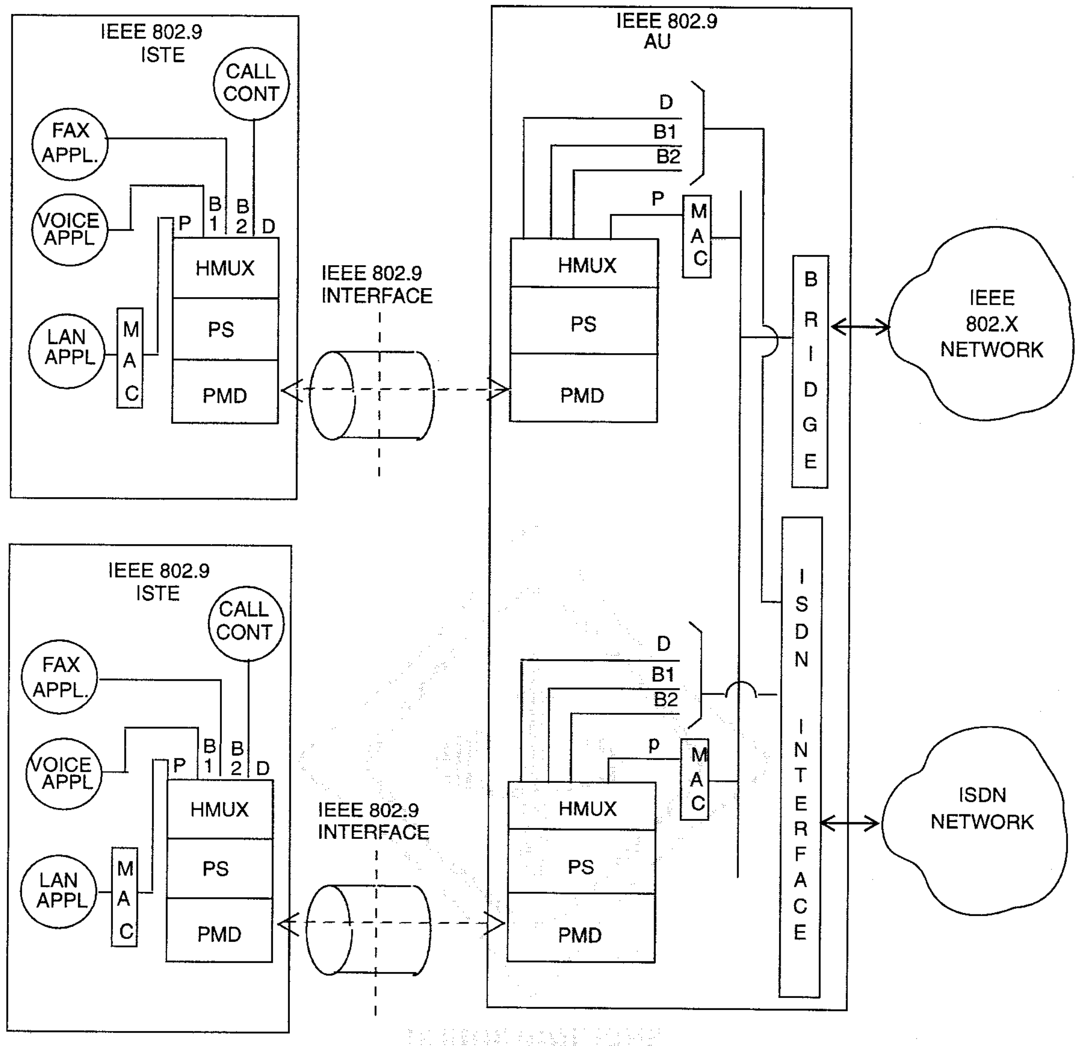


Figure E.3—Bearer service in a Mode 2 environment

E.4 Bearer service in a Mode 3 environment

Figure E.4 illustrates an environment wherein the complete set of ISDN and IEEE 802 LAN services may be simultaneously supported. Within the HMC field of the IEEE 802.9 TDM frame, there is the ability to specify the configuration under which local bandwidth management is be effected between the ISTE and the AU within the IEEE 802.9 management and signalling procedures. The configuration types are defined as "modes" of operation. Figure E.4 depicts a topology with Mode 3 operation. The local call control entity shall use CCITT Q.93x to negotiate the dynamic bandwidth management with the AU. This means that:

a) The ISTE may use the B1 or B2 fixed channel slots.
b) The ISTE may negotiate for the payload to be configured into C channels.
c) The remainder of the payload (not allocated for C channel usage) may be totally devoted to an IEEE 802 LAN communication over the P channel.
d) The support of the ISDN CCITT Q.93x based signalling is essential and shall be used to support the local bandwidth management via CCITT Q.93x Codeset 0 and/or Codeset 7 as well as the CCITT Q.93x flow to the ISDN backbone network (refer to 8.4.4.1.2 for a description of these two separate CCITT Q.93x signalling streams).

In the AU, the involvement of the connection management can be via a simple multiplexer arrangement wherein the D channel messages are simply routed on toward the ISDN. Alternatively, the AU may be part of the ISDN switching network where call requests are processed. In this second situation, the AU can provide local circuit/packet switched connections from one ISTE port to another ISTE port.

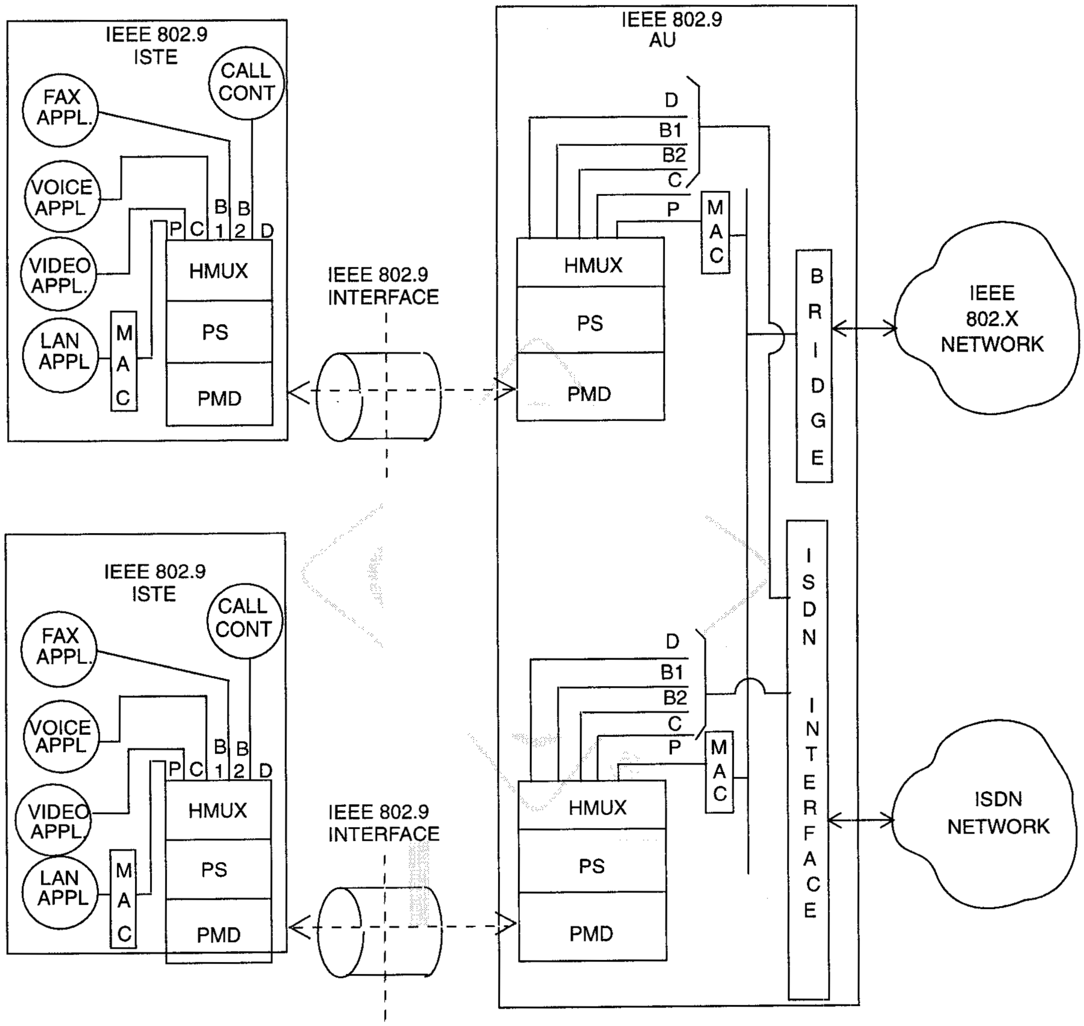


Figure E.4—Bearer service in a Mode 3 environment

E.5 Relationship of IEEE 802.9 subnetwork management entities in AU-to-AU topology

Figures E.5 and E.6 illustrate the topology wherein one AU is connected directly to a second AU. There are two possible arrangements for the role of overall network management:

a) There is a stand-alone central manager
b) One of the AUs serves as the overall manager

In figure E.5, both AUs are equivalent peer devices that report to the central manager. In figure E.6, one of the AUs takes on the role of "slave" and the other AU takes on the role of "master" with respect to execution of network management coordination.

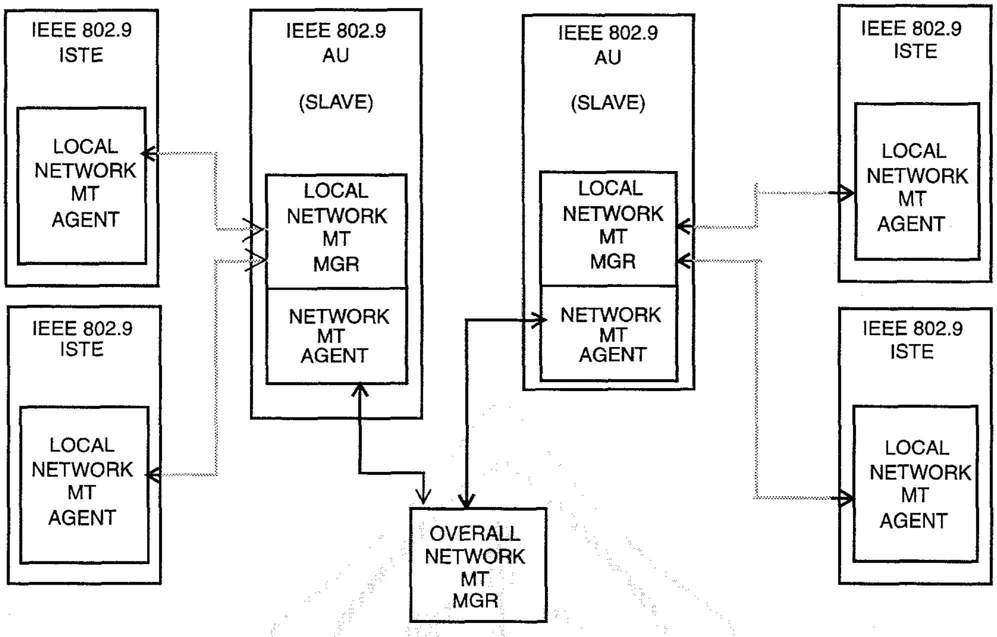


Figure E.5—Relationship of IEEE 802.9 subnetwork management entities with central manager

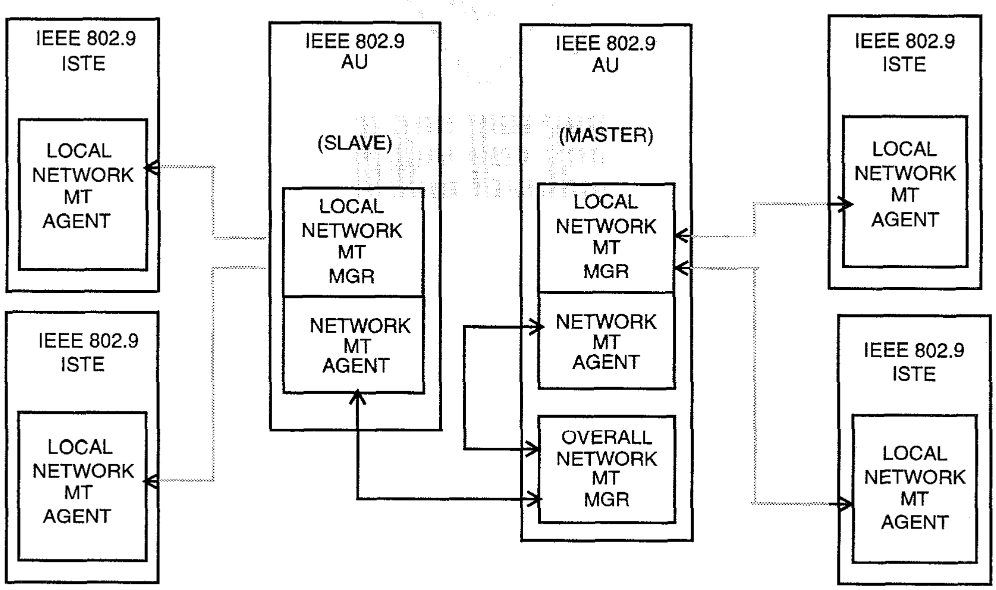


Figure E.6—Relationship of IEEE 802.9 subnetwork management entities with integral manager

Annex F
Sequence and description language (SDL) description of the information flow across layers

(informative)

F.1 Overview

This annex contains a supplemental description to augment the body of this standard. The text description in Chapters 1 through 11 is considered to be the definitive source of description.

This annex contains a description of the signal flow across the layers within the IEEE 802.9 architecture. This signal flow has been represented as an SDL model wherein functional modules, named "processes," exchange communication via "messages." The SDL presentation begins with an illustration of the overall relationship between the user, the ISTE, and the access unit (AU). This presentation progressively "zooms in" to focus on inspecting the ISLAN MAC protocol engine. This illustration technique uses a boldface outlined rectangular box to identify the module that will be examined more closely on a following page.

The progression of functional process inspection is as follows over the following pages:

figure F.1	System ISLAN
figure F.2	Substructure ISTE
figure F.3	Substructure ISTE protocol engine
figure F.4	Substructure ISAU protocol engine
figure F.5	Substructure DLC
figure F.6	Block ISLAN MAC protocol engine

A glossary defining the terms and acronyms used in this SDL description can be found following figure F.6.

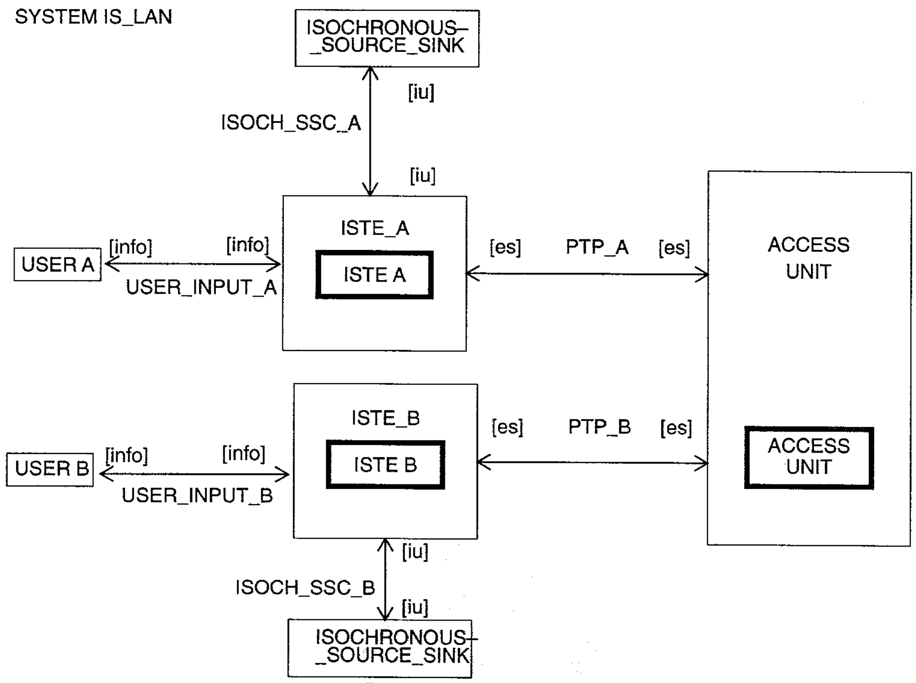


Figure F.1—System ISLAN SDL diagram

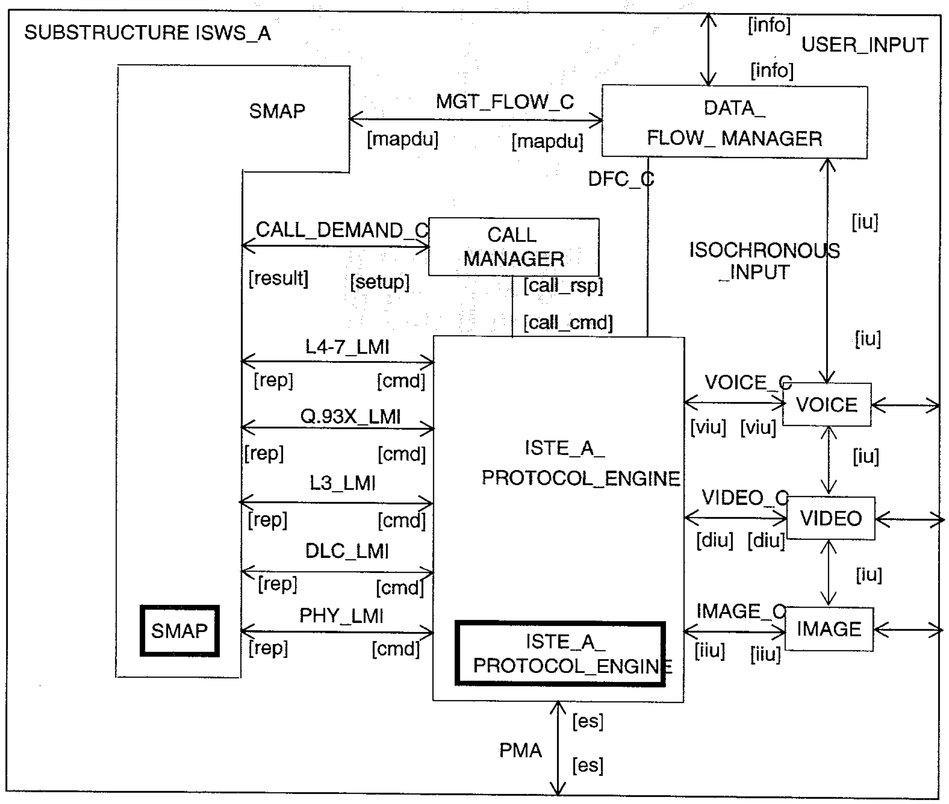


Figure F.2—Substructure ISTE SDL diagram

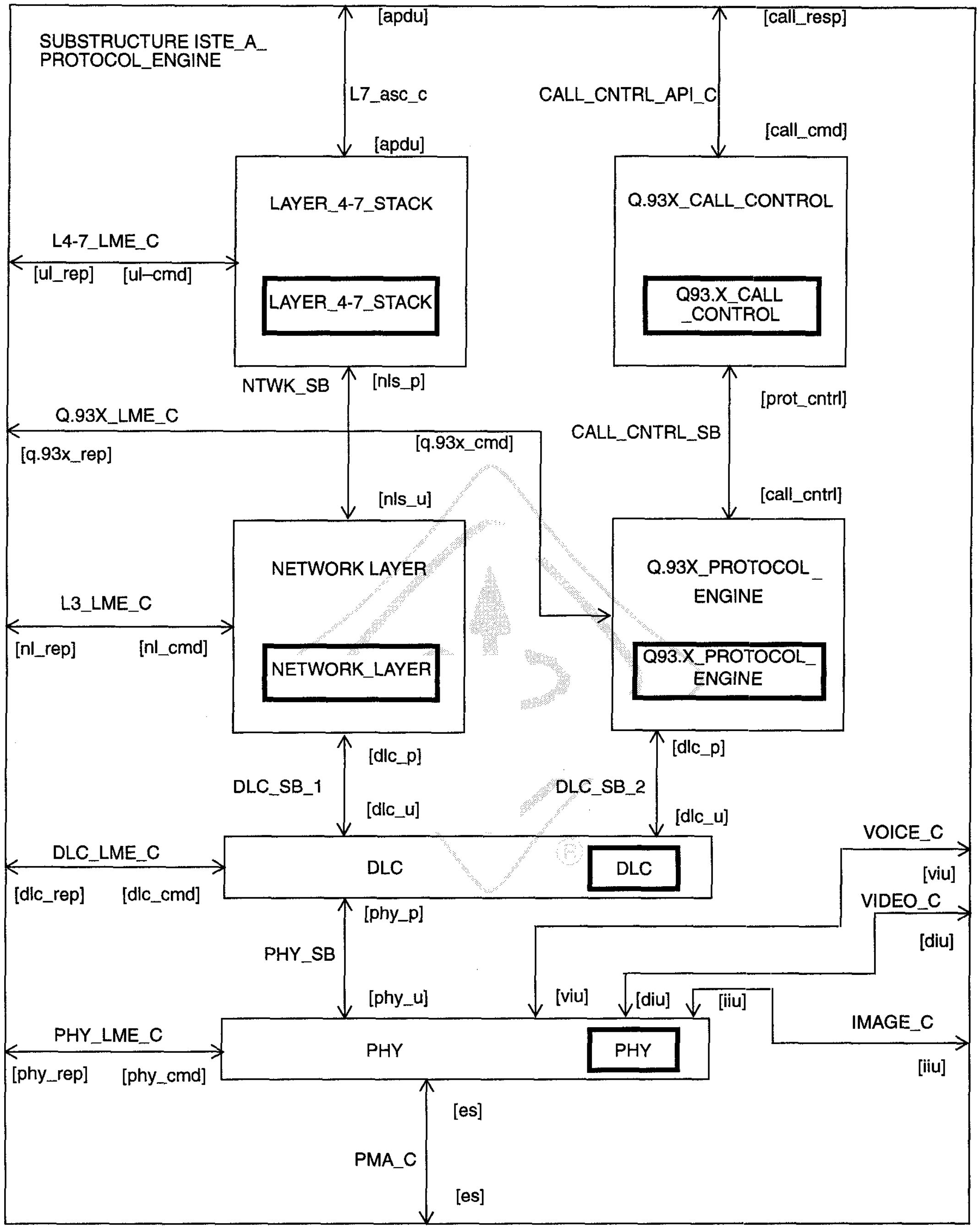


Figure F.3—Substructure ISTE protocol engine SDL diagram

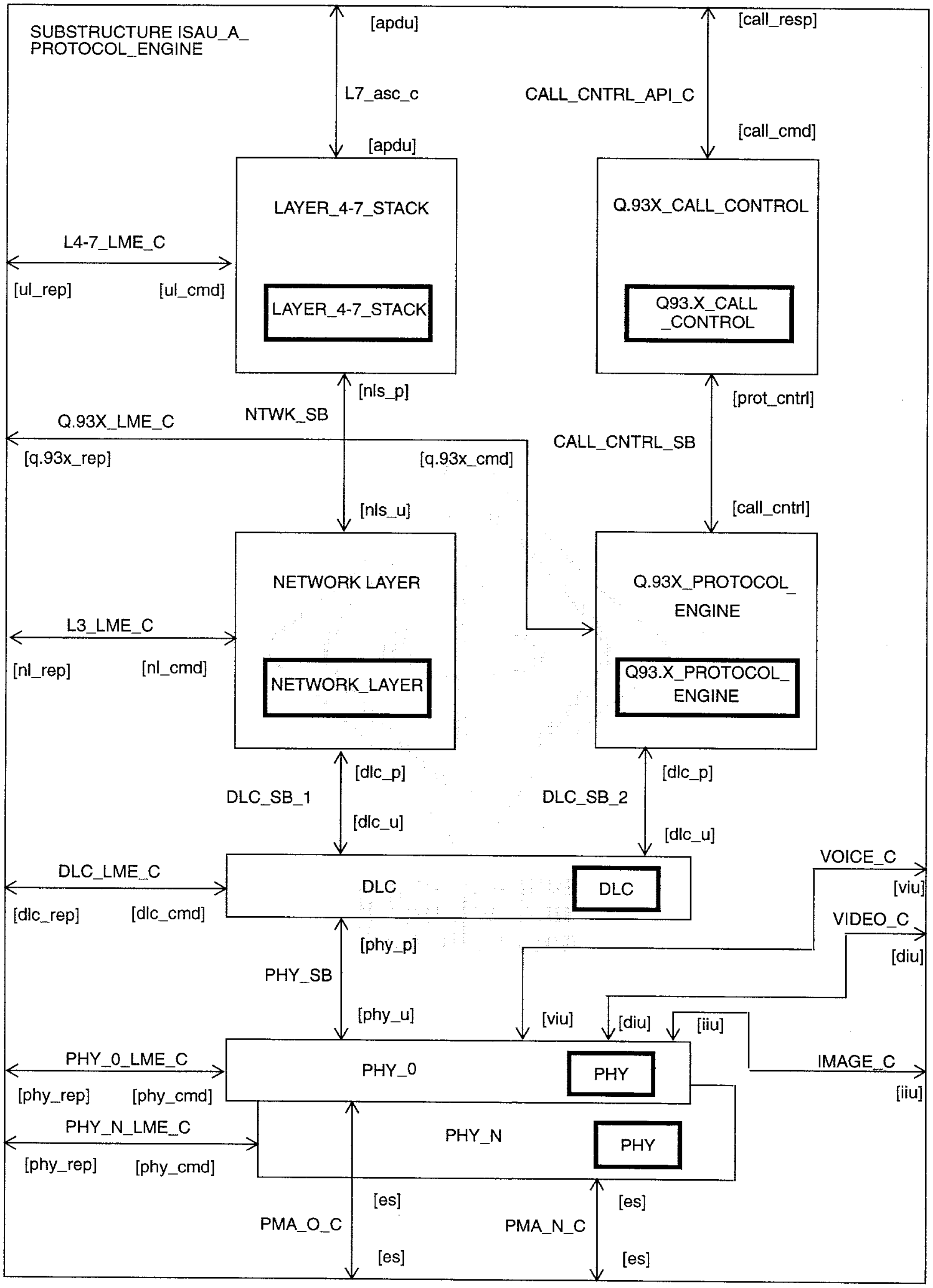


Figure F.4—Substructure ISAU protocol engine SDL diagram

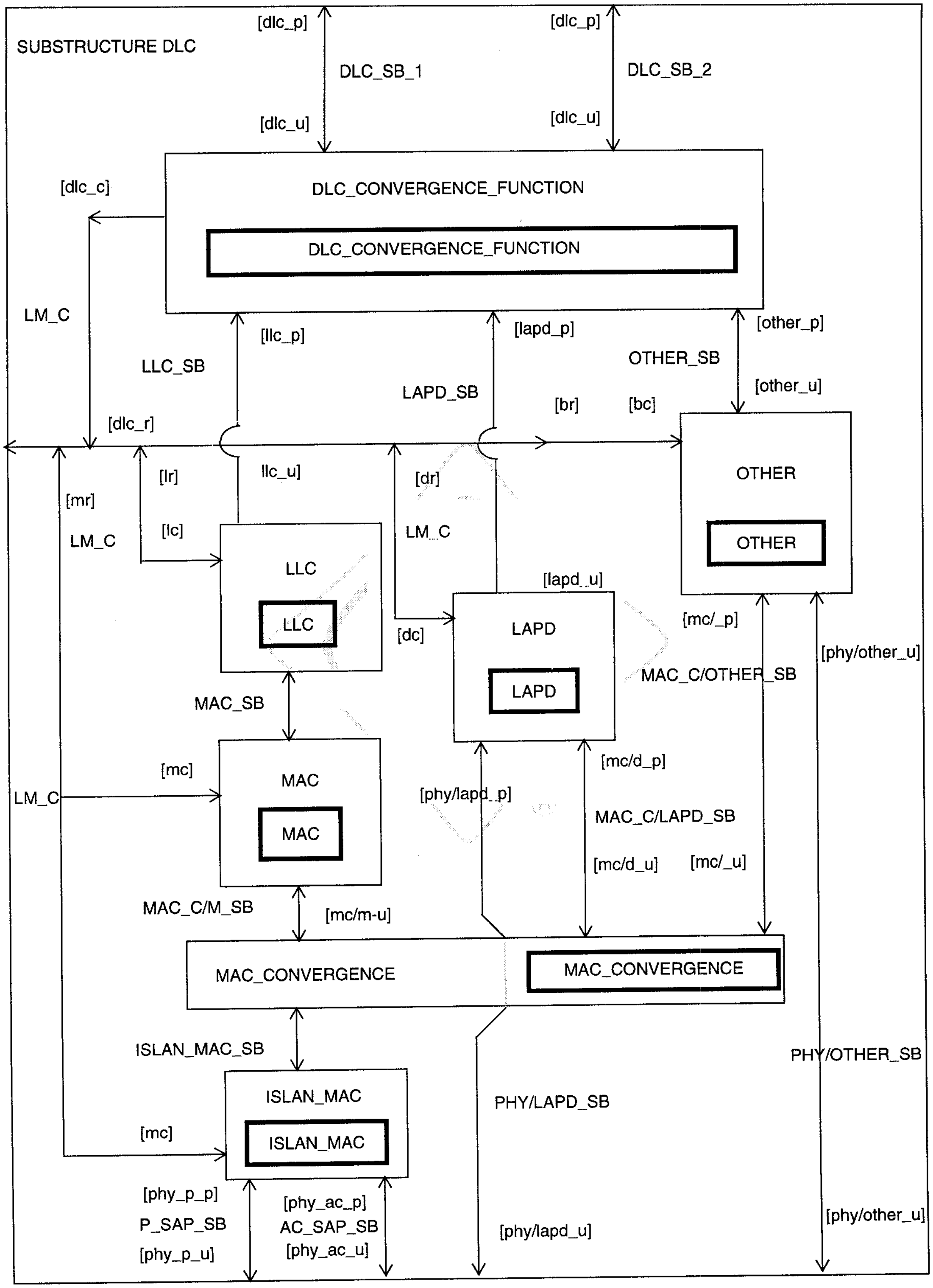


Figure F.5—Substructure DLC SDL diagram

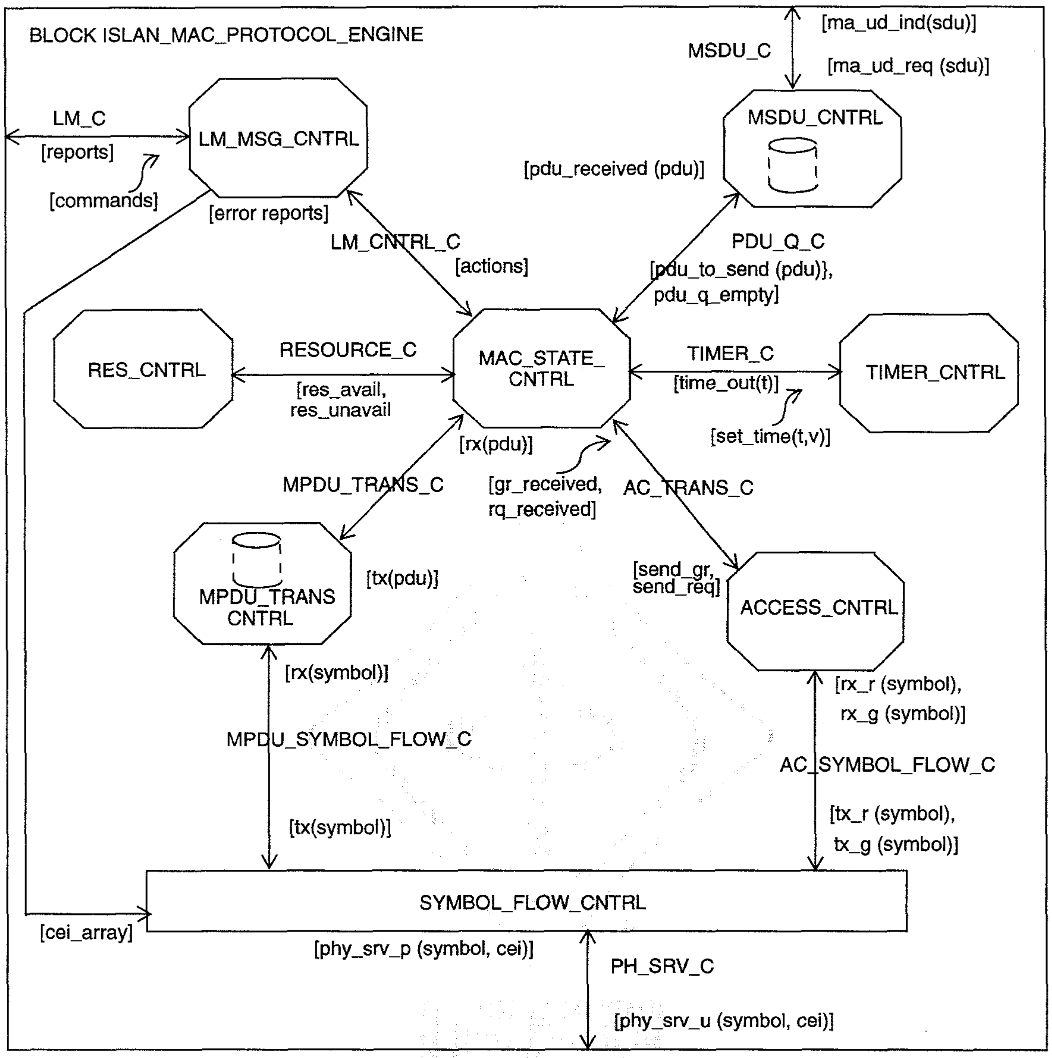


Figure F.6—Block ISLAN MAC protocol engine SDL diagram

F.2 System ISLAN

Figure F.1 illustrates the SDL diagram for the system ISLAN. This highest level SDL diagram depicts the information flow among the ISTEs in an IEEE 802.9 ISLAN.

F.2.1 Block names

ACCESS_UNIT	An IEEE 802.9 access unit (AU).
ISOCHRONOUS_SOURCE_SINK	Devices, external to the workstation, that send and/or receive isochronously transmitted information, e.g., a video camera or a telephone.
ISTE_A	This is a typical ISLAN workstation. There is no architectural significance in the fact that only two are represented.

ISTE_B — This is a typical ISLAN workstation. There is no architectural significance in the fact that only two are represented.

USER — Some end user, i.e., a human operator or a high-level application.

F.2.2 Channel names

ISOCH_SSC_A — Isochronous source/sink A describes a device that is capable of sending and/or receiving isochronous information to the IS workstation.

ISOCH_SCC_B — Isochronous source/sink B describes a device that is capable of sending and/or receiving isochronous information to the IS workstation.

PTP_A — Point-to-point telephone twisted pair (TTP) cable link between the TE and an AU.

PTP_B — Point-to-point TTP cable link between the TE and an AU.

USER_INPUT_A — The user interface for human operators or high-level applications to the workstation.

USER_INPUT_B — The user interface for human operators or high-level applications to the workstation.

F.2.3 Signal names

es — Electrical signal on cable pairs.

info — End-user information submitted for processing, including communication, to the workstation.

iu — Isochronous unit.

F.3 Substructure ISTE_A

Figure F.2 illustrates the SDL diagram for the substructure ISTE_A. This block substructure shows the internal functional organization of an ISTE.

F.3.1 Block names

CALL_MANAGER — The functional unit that commands the CCITT Q.93x signalling functions for bearer service provision.

DATA_FLOW_MANAGER — A source and sink of application data units and information from an operator of the workstation or an external device.

IMAGE — A source/sink, internal to the workstation, of isochronously transmitted information, e.g., a built-in phone.

ISTE_A_PROTOCOL_ ENGINE	The functional unit that is responsible for all OSI and ISDN protocol processing.
SMAP	An OSI system management application process (SMAP) running either as a local agent or a central manager for a distributed OSI management scheme for the ISLAN.
VIDEO	A source/sink, internal to the workstation, of isochronously transmitted information, e.g., a built-in phone.
VOICE	A source/sink, internal to the workstation, of isochronously transmitted information, e.g., a built-in phone.

F.3.2 Channel names

CALL_DEMAND_C	A channel that allows an SMAP to govern call setup, maintenance, and clearing.
CALL_MGT_C	A channel for the control of signalling as a workstation application.
DFC_C	A channel for application data flow.
DLC_LMI	Layer management interface (LMI) channel for the interchange of management information between the SMAP and the layer management entities (LMEs) within the ISTE_A_PROTOCOL_ENGINE.
IMAGE_C	Channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.
ISOCHRONOUS_INPUT	A channel for the flow of isochronously transmitted information from both internal and external sources/sinks to the DATA_FLOW_ MANAGER for incorporation into application protocol data units (APDUs). An example would be high-resolution images captured by a still camera and included in an X.400 interpersonal message.
L3_LMI	LMI channel for the interchange of management information between the SMAP and the LMEs within the ISTE_A_PROTOCOL_ENGINE.
L4-7_LMI	LMI channel for the interchange of management information between the SMAP and the LMEs within the ISTE_A_PROTOCOL_ENGINE.
MGT_FLOW_C	A channel for the exchange of management application protocol data units (MAPDUs), structured in accordance with CMIS/P, between remote SMAPs.
PHY_LMI	LMI channel for the interchange of management information between the SMAP and the LMEs within the ISTE_A_PROTOCOL_ENGINE.
PMA	A physical medium attachment channel for the bidirectional flow of electrical signals, which are encoded for digital line transmission.
Q93x_LMI	LMI channel for the interchange of management information between the SMAP and the LMEs within the ISTE_A_PROTOCOL_ENGINE.

USER_INPUT	A channel for information passing to and coming from external end users and/or devices.
VIDEO_C	A channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.
VOICE_C	A channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.

F.3.3 Signal names

apdu	Layer 7 application protocol data unit.
call_cmd	A call-related command from the CALL_MANAGER.
call_rsp	A call response message.
cmd	A command message across the LMI.
diu	A video isochronous unit.
es	Electrical signal on cable pairs.
iu	Isochronous unit.
iiu	An image isochronous unit.
info	End-user information submitted for processing, including communication, to the workstation.
mapdu	A management application protocol data unit (MAPDU).
rep	A report message across the LMI.
result	A result message for an SMAP-initiated call management operation by the CALL_MANAGER.
setup	An SMAP-initiated call initiation message.
viu	A voice isochronous unit.

F.4 Substructure iste_a_protocol_engine and isau_protocol engine

Figures F.3 and F.4 illustrate the SDL for the substructure ISTE_A protocol engine and the ISAU protocol engine, respectively. These block substructures show the internal protocol stack architectures of the ISTE and the ISAU.

F.4.1 Block names

DLC	Data Link Control.

LAYER_4-7_STACK	OSI Layer 4–7 entities.
NETWORK_LAYER	OSI Layer 3 entity.
PHY	ISTE Physical Layer.
PHY_0	ISAU Physical Layer for the 0th port.
PHY_N	ISAU Physical Layer for the Nth port.
Q.93x_CALL_CONTROL	A functional entity that commands the CCITT Q.93x protocol processing function.
Q.93x_PROTOCOL_ ENGINE	A control plane Layer 3 signalling entity.

F.4.2 Channel names

CALL_CNTRL_API_C	The call control application programming interface channel.
CALL_CNTRL_SB	The call control service boundary provided by the signalling protocol entity.
DLC_SB_1	Data link control (DLC) service boundary.
DLC_SB_2	DLC service boundary.
DLC_LME_C	The channel by which the LMEs exchange management information with the SMAP.
IMAGE_C	A channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.
L3_LME_C	The channel by which the LMEs exchange management information with the SMAP.
L4-7_LME_C	The channel by which the LMEs exchange management information with the SMAP.
L7_ASE_C	A channel for the exchange of APDUs between a Layer 7 application agent and an appropriate application service entity.
NTWK_SB	Network Layer service boundary.
PHY_SB	Physical Layer service boundary.
PHY_0_LME_C	The channel by which the LME for AU port 0 exchanges management information with the SMAP.
PHY_N_LME_C	The channel by which the LME for AU port N exchanges management information with the SMAP.
PMA_C	A physical medium attachment channel for the bidirectional flow of electrical signals, which are encoded for digital line transmission.

PMA_0_C	A physical medium attachment channel for AU port 0 for the bidirectional flow of electrical signals, which are encoded for digital line transmission.
PMA_N_C	A physical medium attachment channel for AU port N for the bidirectional flow of electrical signals, which are encoded for digital line transmission.
Q.93x_LME_C	The channel by which the LMEs exchange management information with the SMAP.
VIDEO_C	A channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.
VOICE_C	A channel for the bidirectional flow of isochronously transmitted information to and from the ISLAN channel interface.

F.4.3 Signal names

apdu	Layer 7 application protocol data unit.
call_cmd	A call-related command from the CALL_MANAGER.
call_cntrl	A call control messages.
call_resp	A call response message.
diu	A video isochronous unit.
dlc_cmd	Report message from the respective LME to the SMAP.
dlc_p	DLC service provider message.
dlc_rep	Report message from the respective LME to the SMAP.
dlc_u	DLC service user message.
es	Electrical signal on cable pairs.
iiu	An image isochronous unit.
nl_cmd	Report message from the respective LME to the SMAP.
nl_rep	Report message from the respective LME to the SMAP.
nls_p	Network Layer service provider message.
nls_u	Network Layer service user message.
phy_cmd	Command message from the SMAP to the respective LME.
phy_p	Physical Layer service provider message.

phy_rep	Report message from the respective LME to the SMAP.
phy_u	Physical Layer service user message.
prot_cntrl	Protocol control message.
q.93x_cmd	Report message from the respective LME to the SMAP.
q.93x_rep	Report message from the respective LME to the SMAP.
ul_cmd	An upper layer command message from the SMAP to the LME.
ul_rep	An upper layer report message from the LME to the SMAP.
viu	A voice isochronous unit.

F.5 Substructure DLC

Figure F.5 illustrates the SDL diagram for the substructure DLC. This block substructure shows the internal organization of the functional entities involved in Layer 2 service provision.

F.5.1 Block names

DLC_CONVERGENCE_FUNCTION	A convergence function that allows Layer 3 entities to access transparently any one of the sets of Layer 2 services available, using any one of the service primitive families supported.
ISLAN MAC	An ISLAN MAC entity.
LAPD	A link access protocol data entity.
LLC	An IEEE 802 Type 1, 2, or 3 logical link control (LLC) entity.
MAC	An entity that builds/processes IEEE 802.9 MAC frames or supporting an ISO/IEC 10039:1991 conformant service.
MAC_CONVERGENCE	A convergence function that allows LAPD and LAPB, and MAC entities to access the service of the ISLAN MAC.
OTHER	Other framed service protocol entities.

F.5.2 Channel names

AC_SAP_SB	The Physical Layer service boundary for the access control (AC) channel user.
DLC_SB_1	DLC service boundary.
DLC_SB_2	DLC service boundary.

ISLAN_MAC_SB	Service boundaries for the IEEE 802 MAC protocol entities supported at the ISLAN interface.
LAPB_SB	Service boundaries for the Layer 2 protocol entities supported at the ISLAN interface.
LAPD_SB	Service boundaries for the Layer 2 protocol entities supported at the ISLAN interface.
LLC_SB	Service boundaries for the Layer 2 protocol entities supported at the ISLAN interface.
LM_C	A set of layer management channels for the bidirectional flow of management messages between Layer 2 LMEs and the SMAP.
MAC_SB	The MAC service boundary.
MAC_C/LAPD_SB	The service boundary between the LAPD module and the MAC convergence function.
MAC_C/M_SB	The service boundary between the IEEE 802.9 MAC and the MAC convergence function.
PHY/OTHER_SB	The Physical Layer service boundary for the other framed service protocol users.
P_SAP_SB	The Physical Layer service boundary for the MAC information bearer channel user.
PHY/LAPD_SB	The Physical Layer service boundary for the LAPD user.

F.5.3 Signal names

ac_sap_p	AC channel service provider.
ac_sap_u	AC channel service user.
dc	SMAP command sent to the LME of the LAPD entity.
dlc_p	DLC service provider message.
dlc_u	DLC service user message.
dr	LME report from the LAPD entity destined for the SMAP.
lapd_p	Message from the LAPD service provider to its user.
lapd_u	Message from the LAPD service user to its provider.
lc	SMAP command sent to the LME of the 802_MAC entity.
802_mac_p	Message from the 802_MAC service provider to its user.

802_mac_u	Message from the 802_MAC service user to its provider.
lr	LME report from the 802_MAC entity destined for the SMAP.
mc	SMAP command sent to the LME of the ISLAN_MAC entity.
mp	Message from the ISLAN_MAC service provider.
mr	LME report from the ISLAN_MAC entity destined for the SMAP.
mu	Message from the ISLAN_MAC service user.
oc	SMAP command sent to the LME of the OTHER service.
or	LME report from the OTHER entity destined for the SMAP.
other_p	Message from the OTHER service provider to its user.
other_u	Message from the OTHER service user to its provider.
phy/o_b	Message from the OTHER service provider.
phy/o_u	Message from the OTHER service user.
phy/stat_p	Message from the OTHER service provider.
phy/lapd_u	Message from the LAPD service user.
phy/p_sap_p	Message from the P_SAP service provider.
phy/p_sap_u	Message from the P_SAP service user.
phy_p	Physical Layer service provider.
phy_u	Physical Layer service user.

F.6 Block islan_mac_protocol_engine

Figure F.6 shows the SDL diagram for the block ISLAN MAC protocol engine. This is the internal structure of the block responsible for MAC protocol execution.

F.6.1 Process names

ACCESS_CNTRL	The process that controls the transmission and reception of request and grant indications, used in the ISLAN medium access protocol.
LM_MSG_CNTRL	The process that controls the flow of layer management messages between the MAC protocol engine and the MAC LME.
MAC_STATE_CNTRL	The process that controls access to the ISLAN PSAP service for the exchange of MPDUs.

MPDU_TRANS_CNTRL	The process that controls the submission/reception of an MPDU to/from the PHY service boundary as a bit stream.
MSDU_CONTROL	The process that manages the exchange of MSDUs with the MAC service user.
RES_CONTRL	The process that manages resources (memory blocks, etc.).
TIMER_CNTRL	The process that controls timers required by the ISLAN MAC protocol operation.

F.6.2 Signal routes

AC_SYMBOL_FLOW_C	A signal route for the exchange of request and grant symbols.
AC_TRANS_C	A signal route for the exchange of messages between the MAC state controller and the request/grant protocol controller. These messages are submissions of request indications (=YES) or grant indications (=YES), respectively, for transmission, or received request and grant values.
LM_C	A signal route for the exchange of management messages between the MAC LME and the processes that execute the ISLAN MAC protocol.
LM_CNTRL_C	A signal route for the exchange of management messages between the LME message controller and the MAC state controller.
MPDU_TRANS_C	A signal route for the sending of messages to/from the MAC state controller from the MPDU reception controller. These messages contain incoming MPDUs.
MSDU_C	A signal route for exchange of MSDUs between the ISLAN_MAC service provider and the MAC service convergence service user.
PDU_Q_C	A signal route for the exchange of messages between the MSDU exchange controller and the MAC state controller regarding the state of the data unit queues.
P_SAP_C	A signal route for the exchange of a P_SAP symbol.
TIMER_C	A signal route for the exchange of messages between the controller responsible for MAC timers and the MAC state controller.

F.6.3 Signal names

actions	Commands submitted to the MAC state controller by the LME.
commands	An action command from the MAC LME, e.g., a command to reset the Finite State Machines (FSMs) of the MAC protocol processor.
error_report	A notification message regarding an event of management significance sent to the MAC LME.

gr_received	Grants (=YES) received from the transmission peer (i.e., an ISTE or the AU) of the MAC.
ma_ud_ind(sdu)	A MAC unit data indication containing an MSDU delivered to the MAC service user.
ma_ud_req(sdu)	An ISLAN MAC unit data request containing an MSDU with a data payload for transmission to a remote peer of the ISLAN MAC service user.
pdu_received	A message stating that a PDU has been received.
pdu_q_empty	A message stating that there are no more messages in the MPDU transmit queue.
pdu_to_send	A message stating that there is at least one message in the MPDU transmit queue.
r(du)	A message that contains a received MPDU to be queued.
reports	Reports of static changes or errors encountered by the MAC state controller, to be sent to the LME.
rq_received	Requests (=YES) received from the transmission peer (i.e., an ISTE or the AU) of the MAC.
rx(symbol)	A message containing a symbol received.
rx_g(symbol)	A message containing a grant symbol received.
rx_r(symbol)	A message containing a request symbol received.
rx(pdu)	A message containing an incoming MPDU.
send_gr	A message requesting the sending of a grant (=YES) to the transmission peer.
send_req	A message requesting the sending of a request (=YES) to the transmission peer.
set_time(t,v)	A message submitted to the ISLAN MAC timer controller requesting that a timer, t, be set to a value, v.
s(du)	A message submitting an MSDU, received from the MAC service user, to the queue controller.
time_out(t)	A message indicating that a timer, t, has timed-out.
tx(symbol)	A message request to transmit a symbol.
tx(pdu)	A message containing an outgoing MPDU.
tx_g(symbol)	A message containing a grant symbol to be sent.
tx_r(symbol)	A message containing a request symbol to be sent.

Annex G
Multimedia security control provision in IEEE 802.9 ISLAN

(informative)

G.1 Overview

The secure multimedia services provision in networks is critical to certain users such as the Department of Defense and other civilian federal agencies in the United States. Increasingly, there is a need to develop this requirement using off-the-shelf components within the framework of Open Systems Interconnection/ Government OSI profile (OSI/GOSIP) compliance. Within a LAN environment, the scope of such standardization of secure service provision is a matter for the specification of the IEEE 802.10 Working Group. Similarly, the scope of ISDN secure service provision is a matter for the specification of the ITU-T and other national standards committees that are responsible for addressing security within the environment of the CCITT Q.93x protocol family. As the IEEE 802.9 ISLAN interface is applied to this customer application environment, there is the requirement that a secure communications device (SCD) be placed on the network side of the access unit (AU). The ISTE would need to be able to communicate with this SCD using a remote secure control (RSC) operation.

Figure G.1 illustrates the architectural configuration for secure services. The remote security operation shall be able to gate on/off the SCD to enable communication to flow to/from the backbone networks. This architecture requires that the SCD be turned ON/OFF prior to invocation of secure service provision either by using IEEE 802.10 or CCITT Q.93x services. Therefore, this RSC operation may be viewed as auxiliary to the security procedures that are communicated at higher layers of the communication protocol. The implementation of secure services is not within the scope of this standard. However, the invocation of the SCD via the RSC will be an optional part of this standard.

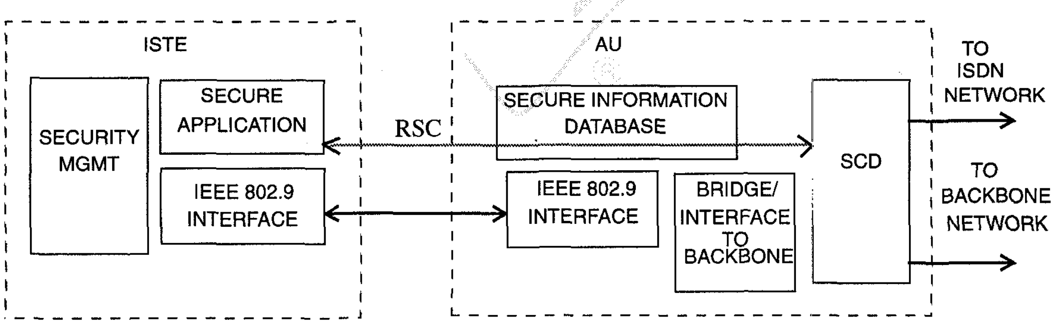


Figure G.1—Architectural configuration for remote secure control of SCD

In figure G.1, the SCD provides an initial secure control procedure for security management to administer the secure services. The procedure involves secure directory initialization for protocol data unit (PDU) exchanges that satisfy the end-to-end security provision. This service has to be administered for both ISDN and backbone data network services. This architecture affords the ISTE with the capability of turning ON/ OFF the security function for connection establishment. The RSC procedure enables the turning on of the SCD prior to the actual secure services being delivered on the network.

The advantage of using IEEE 802.9 services is the fact that the "turn SCD ON or OFF" function is independent of the type of AU, and provides a unified capability to initialize both secure ISDN and secure backbone data services. Once the SCD is turned ON, the secure applications can make use of the ISDN CCITT Q.93x or IEEE 802.10 LAN security protocol or any other proprietary protocol.

G.2 Secure service provision in IEEE 802.9

Figure G.2 illustrates the transport of the RSC information from the ISTE to the SCD (on the network side of the AU). The SCD has been shown as contained within the AU. The application where the SCD is external to the AU is an acceptable model. However, the latter situation would require an external interface specification and therefore is a matter for further study.

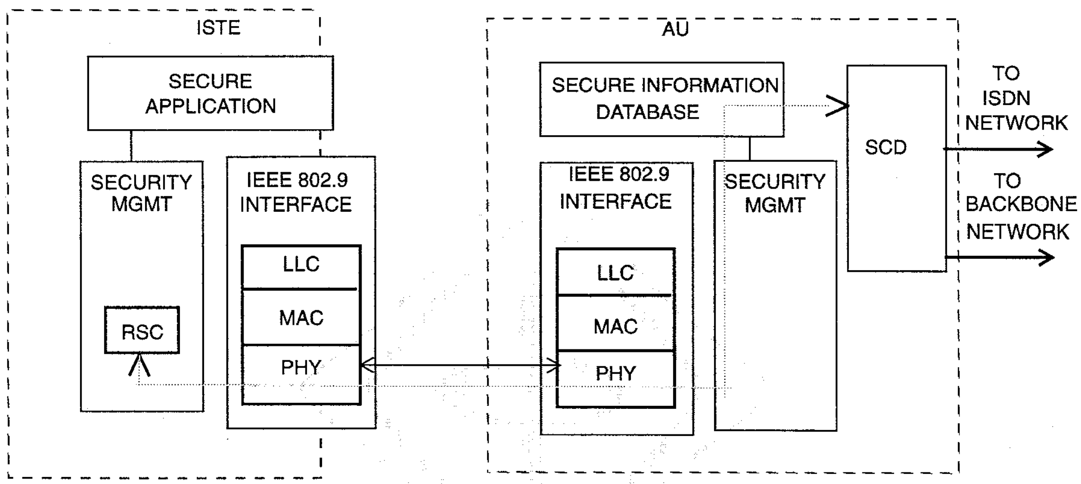


Figure G.2—Transport of remote secure control function across an IEEE 802.9 interface

The security management module within the ISTE is responsible for the initiation of "turning ON" the SCD. The security management entity invokes the process by setting the SECURE control bit (Bit 5) in the access control (AC) field of the IEEE 802.9 TDM frame to a logic "1."

From a service perspective, the security management interface is solely in communication with the Physical (PHY) Layer. The PHY Layer in the ISTE communicates with its peer in the AU to send the control information. Upon reception, the AU security management module will signal the SCD module to turn ON, thus allowing communication to flow toward the backbone network(s). The internal communication from the security management module to the SCD module within the AU is not within the scope of this standard.

The security management module in the ISTE may turn OFF the SCD by sending a request to its PHY Layer that the SECURE control flag be set to logic "0." The AU's management entity, upon reception of the SECURE=0 signal, will act to "turn OFF" the SCD. At this point, communication with the backbone network(s) is halted.

The issue of hysteresis procedures for the peer-to-peer exchange of the SECURE bit is a matter for further study.

It should be noted that while the peer-to-peer exchange of the SECURE bit is a mandatory part of the IEEE 802.9 physical service exchange, the invocation of management procedures to govern the control of this signal is an implementation option.

Annex H
Address interworking across an IEEE 802.9 network

(informative)

H.1 Requirements

This annex covers the addressing requirements and recommendations for the IEEE 802.9 networking system.

To effect the synthesis of the functions provided by IEEE 802 and integrated services digital network (ISDN) standards resulting in implementation of integrated services terminal equipment (ISTE) and networks as provided by the IEEE 802.9 architecture as well as provide access to other types of networks, it is necessary to fully specify how an ISTE utilizes the combined resources of IEEE 802, an ISDN, and/or the other subnetwork. This includes both circuit and packet services.

Ideally, the following universes—IEEE 802.9, the traditional IEEE 802, and the ISDN and/or other subnetworks where applicable—should constitute a universal IS environment. To achieve this goal, a consistent addressing mechanism must be defined among the worlds that the IEEE 802.9 architecture seeks to glue together. It is necessary that an ISTE have the capability to communicate with other ISTEs, with other IEEE 802.x TEs, with ISDN TEs, and with other subnetworks for both circuit and packet services.

Figure H.1 illustrates the environment in which an IEEE 802.9 system may be found. The ISTEs are connected to access units (AUs) which, in turn, provide access to other IEEE 802 local area networks (LANs), ISDNs, or other subnetworks, possibly through interworking units (IWUs). With respect to other types of subnetworks, two classifications may be defined: those that conform to the ISO Network Layer addressing scheme, and those that do not. To meet the above objectives, the following requirements shall be satisfied:

a) *Intra-AU communication.* An ISTE on AU_k shall be able to communicate with another ISTE on AU_k to utilize both IEEE 802 medium access control (MAC) services and ISDN services.
b) *Inter-AU communication.* An ISTE on AU_k shall be able to communicate with another ISTE on AU_l ($k \neq l$) to utilize both IEEE 802 MAC services and ISDN services.
c) *ISTE to other IEEE 802 TEs.* An ISTE shall be able to communicate with another IEEE 802.x ($x \neq 9$) TE to utilize IEEE 802 MAC services.
d) *ISTE to ISDN TE.* An ISTE shall be able to communicate with another TE connected to an ISDN to utilize ISDN services.
e) *ISTE to other TE.* An ISTE shall be able to communicate with another TE connected to another type of subnetwork such as ISO connectionless network protocol (CLNP) or CCITT X.25, which utilizes the ISO Network Layer addressing scheme (ISO/IEC 8348:1993 Addendum 2), or widely available private networks, such as internet protocol (IP) and internet protocol X (IPX), which do not utilize the ISO addressing scheme.
f) *IWUs.* Where IWUs are utilized, they should operate at as low a layer as possible to minimize cost and increase performance.

Since the ISTE of IEEE 802.9 is a member of the IEEE 802 family, it shall have a MAC address, as presently defined by the IEEE 802.9 architecture. This is a basic assumption. Also, it is assumed that any resulting scheme used to satisfy the above requirements will have to conform to existing international or open standards to ensure its acceptance.

Requirement a) above is met through the use of the existing MAC address. In addition to the MAC address, the satisfaction of requirement b) may imply the need for a MAC bridge or a relay function in the AU. This is discussed in annex E. Meeting requirement c) implies an IEEE 802.1-type MAC bridge. Requirement d) imposes a new constraint: interworking with TEs that are addressable at the Network Layer. This also has implications for requirement e). Specifically, the ISDN TEs shall be addressed using the ISDN numbering plan [see CCITT Recommendation E.164 (1991)]. This ensures that it will not be necessary to seek any modification to existing standards. Requirement f) may similarly imply addressing at the Network Layer using the addressing scheme of the other subnetwork. For example, in the case of a CCITT X.25 subnetwork, this would be the CCITT X.121 addressing scheme, or for an IP subnetwork, the IP addressing scheme.

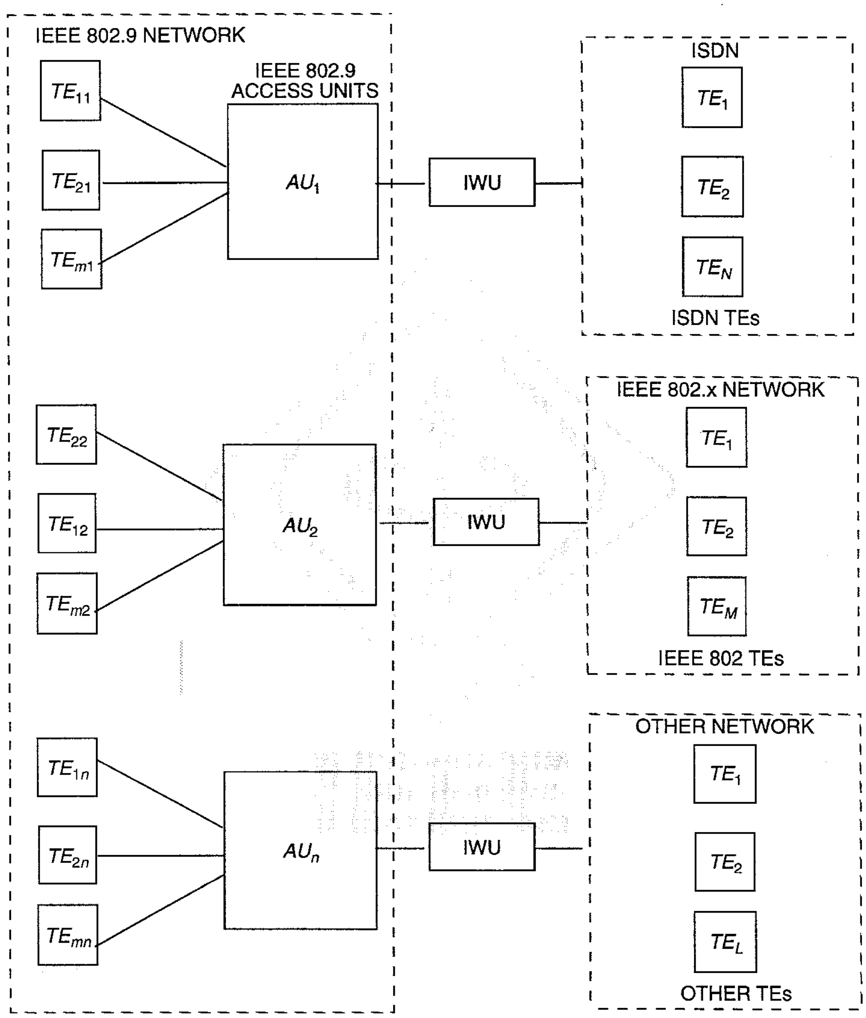


Figure H.1—IEEE 802.9 networking environment

The scheme for the ISTE to maintain its MAC identity while also communicating transparently with the ISDN and other TEs is based on the OSI Network Layer addressing scheme as proposed in ISO/IEC 8348:1993 Addendum 2, the ISDN numbering plan as enunciated in CCITT Recommendation E.164 (1991) with its conveyance implementation in CCITT Recommendation Q.931 (1993), and the IEEE 802 MAC addressing scheme. These are reviewed in H.2, prior to the detailed description of the IEEE 802.9 addressing scheme.

H.2 Relevant addressing schemes

H.2.1 OSI network layer addressing

H.2.1.1 Principles

OSI Network Layer addressing is defined in ISO/IEC 8348:1993 Addendum 2 and defines a scheme for network service access points (NSAPs), the points at which the communications capability of the Network Layer is made available at the layer boundary to users. The NSAP address is the information that the OSI network service provider needs to identify a particular NSAP. Thus, two OSI entities utilizing the services of the Network Layer for communication would access each other via their respective NSAPs using their respective NSAP addresses.

The set of all of the NSAP addresses in the OSI environment constitutes the global network addressing domain. Under the auspices of the global network addressing domain, any OSI network service user can communicate with any other OSI network service user. The NSAP address has the following important characteristics:

- An NSAP address can be defined to unambiguously identify any NSAP.
- At any NSAP, it is possible to identify any other NSAP within any OSI end system.
- The Network Layer protocols established between correspondent network entities convey the complete semantics of an NSAP address, e.g., CCITT Q.931 conveys the semantics of CCITT E.164 and CCITT X.25 conveys the semantics of CCITT X.121.
- An NSAP address always identifies the same NSAP, regardless of which network service user enunciates the address.
- A Network Service user, when given an NSAP address by the network service provider in an indication service primitive, may subsequently use that NSAP address in another instance of communication with the corresponding NSAP.

The global network addressing domain, which is under the joint authority of ISO and CCITT, in turn, may be subdivided into network addressing domains. This results in a hierarchy of addressing domains as shown in figure H.2. The domains have the following properties:

- Any network addressing domain is a subset of the global network addressing domain and consists of all the NSAP addresses that are assigned by one or more addressing authorities.
- Every NSAP address is part of a network addressing domain that is administered by one and only one authority.
- If that network addressing domain is part of a higher addressing domain, the authority for the lower domain is authorized by the authority for the higher domain to assign NSAP addresses from the lower domain.
- Each domain may be further partitioned into subdomains as shown in figure H.2.

ISO/IEC 8348:1993 Addendum 2 defines the first three levels of domains of NSAP addresses. Other levels may be defined, as needed, by the subauthorities under the higher levels, even though the procedures are not completely specified. The second-level domains, the subdomains of the global network addressing domain, are shown in figure H.3.

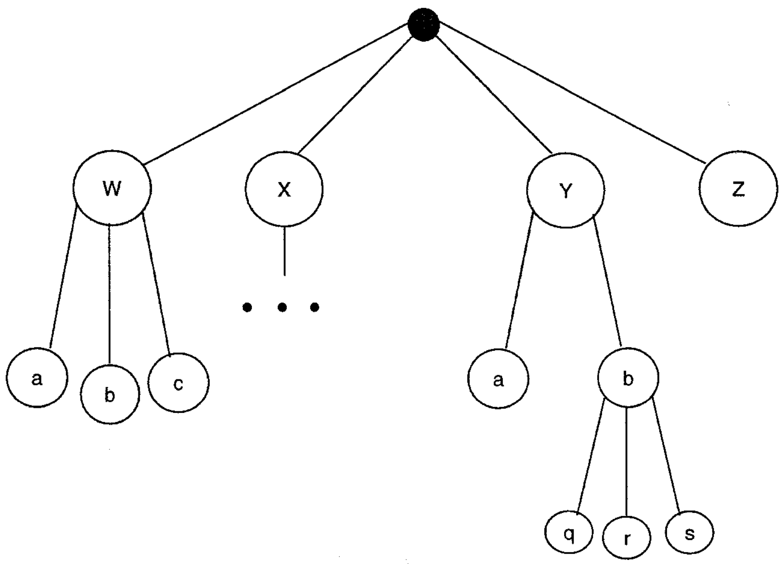


Figure H.2—Hierarchical structure of NSAP addresses

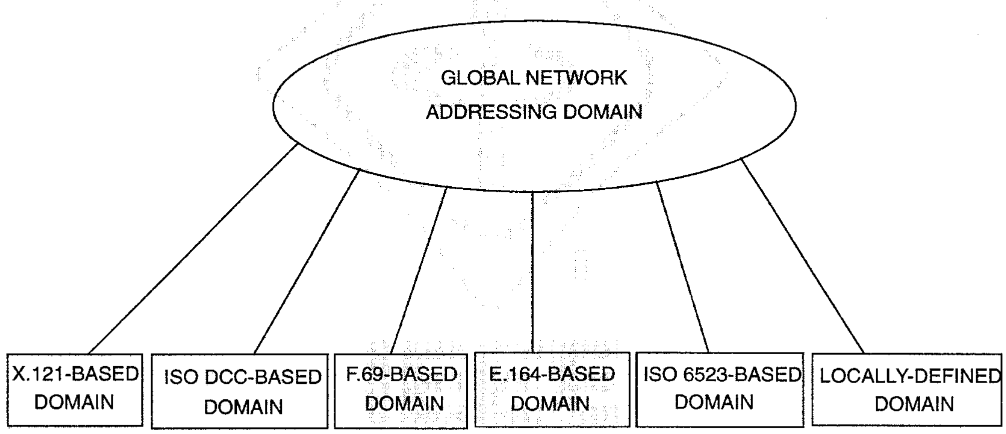


Figure H.3—Global network addressing domain and its subdomains

They are classified as follows:

— A set of four domains, under the auspices of the CCITT, each of which corresponds to a type of public telecommunication network: CCITT Recommendations: X.121 (1992) for X.25 networks, F.69 (1993) for telex networks, and E.164 (1991) for public switched telephone networks and ISDNs.
— Two domains under the auspices of ISO: one based on the ISO data country code (DCC), defined by ISO 3166:1993, assigned to the ISO member organization in each country or an appropriately sponsored organization within the country, and another based on ISO 6523:1984, the international code designator (ICD) identifying various international organizations.
— A local domain, which serves a variety of purposes: the definition of locally-defined NSAPs, facilitation of migration to OSI, and provision for future definition of other domain authorities at the second tier.

Under each of the second-level domains are third-level domains. For example, under each of the domains corresponding to CCITT numbering plans, an individual subnetwork address from the numbering plan corresponds to a set or domain of NSAP addresses based on that subnetwork number. The authority for that subdomain of NSAP addresses is the entity located at the specified subnetwork point of attachment (SNPA) identified by the subnetwork number. For example, for a LAN connected to a CCITT X.25 packet data network (PDN) through an IWU (as shown in figure H.4), the CCITT X.121 address of the PDN would belong to the second-level domain. In particular, the IWU would be assigned a CCITT X.121 address from that domain. That CCITT X.121 address could, in turn, be used to define a set of NSAPs for the third-level domain. A station connected to the LAN could be given an NSAP that is a combination of the CCITT X.121 address and its MAC address, as shown in figure H.4. In this example, the authority of the third-level domain would be the owner of the IWU.

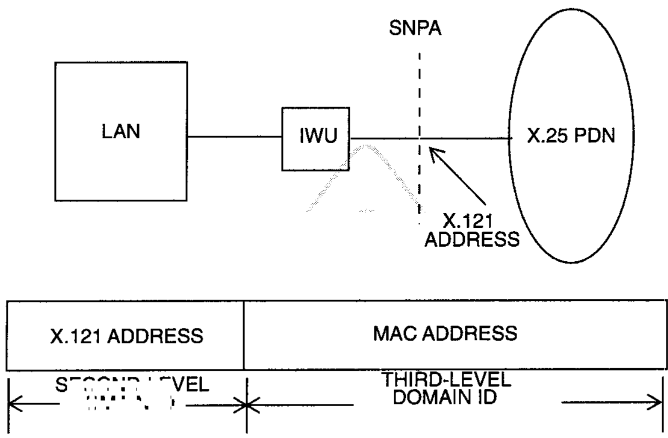


Figure H.4—Example: CCITT X.121 second-level domain with third-level domain

H.2.1.2 Semantics

The OSI Network Layer addressing scheme follows the principle that, at any level of the hierarchy, an initial part of the address unambiguously identifies a subdomain, and the rest is allocated by the authority associated with the subdomain to unambiguously identify either a lower-level subdomain or an NSAP within the subdomain. The NSAP address consists of two basic semantic parts: the initial domain part (IDP) and the domain specific part (DSP). This is illustrated in figure H.5.

The IDP is a network addressing domain identifier and the DSP is the corresponding subdomain address. The IDP specifies a subdomain of the global network addressing domain and identifies the network addressing authority responsible for assigning NSAP addresses in the specified domain. It consists of two parts: the authority and format identifier (AFI) and the initial domain identifier (IDI).

The AFI specifies

— The format of the IDI.
— The network addressing authority responsible for allocating values of the IDI, whether or not leading zero digits in the IDI are significant.
— The abstract syntax of the DSP.

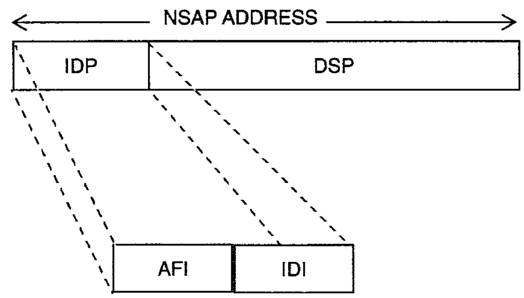


Figure H.5—NSAP Address structure

The IDI specifies

— The network addressing domain from which the values of the DSP are allocated.
— The network addressing authority responsible for allocating values of the DSP from that domain (the semantics of the DSP are determined by the network addressing authority identified by the IDI).

Table H.1 gives the AFI allocations. Among the currently allocated values are those reserved for new IDI formats by ISO and CCITT as well as those reserved for future allocation. The syntax of the IDP is in decimal digits. The AFI consists of an integer value between 0 and 99, with the allocation ensuring that the first decimal digit of the IDP can never be zero. This provides an escape mechanism for use by protocols that expect to hold incomplete NSAP addresses in a field that normally carries a complete NSAP address.

Table H.1—AFI allocations

00-09	Reserved—Will not be allocated
10-35	Reserved for future allocation
36-59	Allocated and assigned to existing IDI formats
60-69	Allocated for assignments to new IDI formats by ISO
70-79	Allocated for assignment to new IDI formats by CCITT
80-99	Reserved for future allocations by joint agreement of ISO and CCITT

A specific combination of IDI format and DSP abstract syntax is associated with each allocated AFI value, as summarized in table H.2. Two AFI values are associated with each combination that involves a variable-length IDI format. In each case, both of the AFI values identify the same combination of IDI format and DSP abstract syntax. The numerically lower AFI value is used when the first significant digit in the IDI is nonzero and the numerically greater AFI value is used when the first significant digit in the IDI is zero. This artifice allows the distinction between when leading zeros are part of the IDI and when they are merely used for padding, when the IDI field is not full.

The IDI format types correspond to the second-level domains defined in figure H.3, namely,

— CCITT X.121 format consisting of up to a 14-digit numeric code
— ISO DCC IDI format consisting of a 3-digit numeric code

- CCITT F.69 format consisting of up to 8 digits commencing with a 2- or 3-digit destination code
- CCITT E.164 format with up to 15 digits commencing with the country code
- ISO ICD format consisting of a 4-digit ICD identifying an international organization
- Local IDI format where the IDI is null and the DSP is allocated according to local considerations

Since the IDI constitutes the second-level domain and the DSP constitutes the third-level domain, values of the DSP are allocated by the network addressing authority identified by the IDI. Following the example above, the IDI using the CCITT X.121 format is the second-level domain. Thus, the DSP, which is the third-level domain, is allocated by the owner of the IWU who, in this case, has chosen to use MAC addressing.

Table H.2—Allocated AFI values

IDI format	DSP Syntax			
	Decimal	Binary	Character	National character
CCITT X.121	36, 52	37, 53	—	—
ISO 3166 DCC	38	39	—	—
CCITT F.69	40, 54	41, 55	—	—
CCITT E.164[a]	44, 48	45, 59	—	—
ISO 6523 ICD	46	47	—	—
Local	48	49	50	51

[a] CCITT Recommendation E.164 replaces E.163.

H.2.1.3 Encoding and conveyance of the NSAP

ISO/IEC 8348:1993 Addendum 2 specifies the abstract syntax of the NSAP address but leaves the specification of the particular encodings of the NSAP address to the particular Network Layer protocol. Nevertheless, it does define two preferred encodings, namely,

- The preferred decimal encoding, in which the entire NSAP address is represented as a string of decimal digits
- The preferred binary encoding, in which the entire NSAP address is represented as a string of binary digits

In both of these encodings, the representation of both the AFI and the IDI consists of a fixed number of digits or octets (and hence the need for padding), while the representation of the DSP can be of variable length.

The maximum length is specified in terms of the preferred encodings: 40 digits for the preferred decimal encodings and 20 octets for the preferred binary encoding. Thus, a protocol that conforms to ISO/IEC 8348:1993 Addendum 2 shall be capable of conveying up to 40 decimal digits or 20 binary octets per address. This is one of the requirements for the provision of the Network Layer service. Network Layer protocols with the capability to convey the NSAP include the OSI connectionless protocol, the 1993 version of the CCITT X.25 protocol for the provision of connection-mode service, and CCITT Q.931. Note that in 1984 when CCITT X.25 was enhanced to enable it to fully support the OSI Network Layer service, a major modification was the introduction of the called and calling address extension facilities to permit the carrying of full NSAP addresses.

Corresponding to the maximum length NSAPs, table H.3 shows the corresponding maximum DSP lengths. This places a limitation on the third-level domain address length.

Table H.3—Maximum DSP length

IDI format	DSP syntax			
	Decimal	Binary	Character	National character
CCITT X.121	24	9	—	—
ISO 3166 DCC	35	14	—	—
CCITT F.69	30	12	—	—
CCITT E.164[a]	23	9	—	—
ISO 6523 ICD	34	13	—	—
Local	48	15	19	7

[a] CCITT Recommendation E.164 replaces E.163.

H.2.2 ISDN addressing

The ISDN address structure is provided in CCITT Recommendation E.164 (1991) and the procedure for conveying the address is provided in CCITT Recommendation Q.931 (1993). CCITT Recommendation E.164 (1991) defines an IDI format of variable length up to 15 decimal digits, with a provision for a DSP. The 15 digits are composed of a country code (CC) and a national significant number (NSN), which, in turn, may be subdivided into a national destination code (NDC) and a subscriber number (SN). This is illustrated in figure H.6.

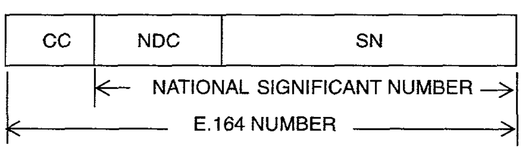


Figure H.6—Structure of CCITT E.164 address

The CC is a variable field of up to 3 digits used to select the destination country. The NSN is used to select the destination subscriber. The NDC will depend upon the needs of the country and may describe such things as destination subnetworks within the country, trunk code, etc. This composition may be viewed as a way of defining subdomains of the IDI enabling CCITT to define subauthorities in the administration of CCITT E.164. For example, under a particular country code, the Post, Telegraph and Telecommunications (PTT) administration of the country is given the authority to administer the NSN. Through the use of the NDC field, the PTT could, in turn, define its own subadministrators.

CCITT Recommendation E.164 also makes provision for subaddressing. While this capacity is outside of the ISDN numbering plan, it is, nevertheless, an intrinsic part of the ISDN addressing capabilities. It makes possible the identification of an entity within a subscriber's subnetwork of a point beyond the SNPA, i.e., that defined by the ISDN number. As explained below, CCITT Q.931 provides the mechanism for the transfer of the address information from the ISDN to the SNPA. This is illustrated in figure H.7. The CCITT E.164 subaddress accommodates up to 40 decimal digits or 20 octets and is transferred to the equipment being addressed by the CCITT E.164 address. When required, the subaddress is sent by the calling party within the call setup procedure and is passed transparently through the network as a separate entity from both the ISDN number and the user-to-user information. It is not required to be processed by the public ISDN.

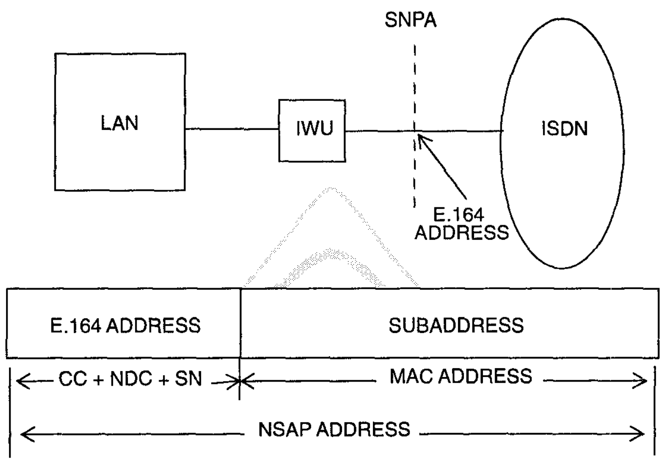


Figure H.7—Conveying MAC address using CCITT Q.931

For the conveyance of the called and calling party numbers and subaddresses, CCITT Q.931 defines information elements (IEs). The formats are shown in figures H.8 and H.9, respectively, for the called and calling party numbers and figures H.10 and H.11, respectively, for the called and calling party subaddresses. The first octet of the IE is the IE identifier for called/calling party number or subaddress. The corresponding code designations are summarized in table H.4.

BITS 8	7	6	5	4	3	2	1	OCTETS
	CALLED PARTY NUMBER							
0	1	1	1	0	0	0	0	1
	INFORMATION ELEMENT IDENTIFIER							
LENGTH OF CALLED PARTY NUMBER CONTENTS								2
1 EXT	TYPE OF NUMBER			NUMBERING PLAN IDENTIFICATION				3
0	NUMBER INFORMATION							4 ETC.

Figure H.8—Format of the called party number IE in CCITT Q.931

BITS 8	7	6	5	4	3	2	1	OCTETS
0	CALLING PARTY NUMBER 1	1	0	1	1	0	0 INFORMATION ELEMENT IDENTIFIER	1
LENGTH OF CALLING PARTY NUMBER CONTENTS								2
1 EXT	TYPE OF NUMBER			NUMBERING PLAN IDENTIFICATION				3
0	NUMBER INFORMATION							4 ETC.

Figure H.9—Format of the calling party number IE in CCITT Q.931

BITS 8	7	6	5	4	3	2	1	OCTETS
0	CALLED PARTY SUBADDRESS 1	1	1	0	0	0	0 INFORMATION ELEMENT IDENTIFIER	1
LENGTH OF CALLED PARTY SUBADDRESS CONTENTS								2
1 EXT	TYPE OF SUBADDRESS		ODD/EVEN INDICATOR	0 SPARE	0	0		3
SUBADDRESS INFORMATION								4 ETC.

Figure H.10—Format of the called party subaddress IE in CCITT Q.931

BITS 8	7	6	5	4	3	2	1	OCTETS
0	CALLING PARTY SUBADDRESS 1	1	1	0	0	0	0 INFORMATION ELEMENT IDENTIFIER	1
LENGTH OF CALLING PARTY SUBADDRESS CONTENTS								2
1 EXT	TYPE OF SUBADDRESS		ODD/EVEN INDICATOR	0 SPARE	0	0		3
SUBADDRESS INFORMATION								4 ETC.

Figure H.11—Format of the calling party subaddress IE in CCITT Q.931

Table H.4—IE identifiers

0000111	Called party number
0011011	Calling party number
1000111	Called party subaddress
1011011	Calling party subaddress

The second octet of each IE gives the length of the IE. The third octet of the called/calling party number provides for the extension bit, the number type, and the numbering plan identification while the third octet of the called/calling party subaddress provides for the extension bit, the type of subaddress, an odd/even field indicator specifying even or odd number of address signals [used when user specified and the coding is binary coded decimal (BCD)], and some spare bits that are set to "0."

The type of number and numbering plan identification are shown in tables H.5 and H.6. By combining the "network specific" type and "private numbering plan" identification, it is possible to use CCITT Q.931 for the conveyance of numbering plans other than CCITT E.164 and other international standard numbering plans. In particular, it is possible to convey addresses such as IP, IPX, XNS, Netbios®, and Appletalk® that are commonly found in private networks. For the private network application, octet 4, the first octet of the number information, may be used to designate the type of addressing being used. Table H.7 gives code designations for some of the well-known private network addressing plans.

Table H.5—Type of number

Bits 765	Designation
000	Unknown
001	International number
010	National number
011	Network-specific number
100	Subscriber number
110	Abbreviated number
111	Reserved for extension
All other values	Reserved

Table H.6—Numbering plan identification

BIts 4321	Designation
0000	Unknown
0001	ISDN/telephony numbering plan
0011	Data numbering plan (CCITT X.121)
0100	Telex numbering plan (CCITT F.69)
1000	National standard numbering
1001	Private numbering plan
1111	Reserved for extension
All other values	Reserved

The subaddress type field indicates whether the subaddress is an NSAP (as defined by ISO/IEC 8348:1993 Addendum 2), user specified, or reserved. The definitions of the subaddress type field and the odd/even indicator bit are summarized in tables H.8 and H.9, respectively.

Table H.7—Address field identification

Address type (Hexadecimal)	Designation
0010	IEEE MAC address Reference: IEEE Std 802-1990
0021	Netbios® individual name Reference: IBM® LAN Technical Reference (SC30-3383)
0030	IP Internet address Reference: Address Mappings-RFC 796
0032	IPX address Reference: Novell, Inc. Advanced Netware, V2.0 Internet Exchange Protocol (IPX)
0034	XNS address Reference: Xerox Corporation Internet Standard Protocol XNSS-028112
0036	Appletalk® address
0060	NSAP Reference ISO/IEC 8348:1993 Addendum 2

Table H.8—Type of subaddress

Bits 765	Designation
000	NSAP (CCITT X.213, ISO/IEC 8348:1993 Addendum 2)
010	User specified
All other values	Reserved

Table H.9—Odd/even indicator

Bit 4	Designation
0	Even number of address signals
1	Odd number of address signals

The subaddress information starts at octet 4. This is formatted according to the type: ISO/IEC 8348:1993 Addendum 2 or user specified. Where the called/calling party number designates the IDI of an addressing plan conforming to ISO/IEC 8348:1993 Addendum 2, the subaddress may be used to carry the DSP. For example, if CCITT E.164 or CCITT X.121 is used as the plan for the called/calling party number, then the called/calling party subaddress could be a MAC address. Similarly, if the called/calling party number, through the "network specific" option is used, then the address spaces could be expanded by augmenting them by the MAC address.

Thus, through the combined network-specific number/private numbering plan designation or the user-specified feature, CCITT Q.931 can be used to convey addresses other than NSAPs. This makes it possible for an ISTE to utilize both public network and private network addresses.

H.2.3 IEEE 802 MAC addressing

H.2.3.1 Principles

The IEEE 802 MAC address provides a mechanism for IEEE 802 TEs to communicate at the MAC sublayer. It consists of 6 octets. IEEE, under the authority of ANSI, administers the addresses. IEEE, in turn, gives blocks of addresses to other organizations that manufacture the MAC entities. The individual addresses are universally unique.

The IEEE 802 MAC address is not a Network Layer address and so cannot constitute an NSAP in the context of ISO/IEC 8348:1993 Addendum 2. However, it may be considered a DSP. Thus, it could be combined with one of the IDPs (AFI and IDI) to form an NSAP. In that form, since only the MAC address is recognizable by the MAC entity, when a full NSAP is used, the MAC address must be derived from it before being passed on to the MAC entity. Such a function could be performed by an IWU. For example, for a LAN attached to an ISDN, the IWU could take the NSAP and translate to the appropriate MAC address. This translated NSAP is conveyed in the address fields of the MAC protocol data unit (MPDU).

H.3 IEEE 802.9 addressing

H.3.1 Principles

The following principles, based on the OSI Network Layer addressing, ISDN addressing, and IEEE 802 MAC addressing, in conjunction with CCITT Q.931 address conveyance, underlie the IEEE 802.9 addressing scheme.

— For an ISTE to communicate with an ISDN or utilize the ISDN services of IEEE 802.9, it shall employ a CCITT E.164 address. Similarly, to communicate with any other TE on a subnetwork that conforms to ISO/IEC 8348:1993 Addendum 2, it shall utilize that subnetwork's address. Where the subnetwork utilizes a private addressing scheme, that private network plan will need to be used for communication.

— The CCITT E.164 address constitutes a valid IDI of the NSAP for the ISTE when it provides the Layer 3 service, as on the D channel; the same is true for addressing schemes, such as CCITT X.121, that conform to ISO/IEC 8348:1993 Addendum 2.

— The IEEE 802 MAC address may be considered the DSP of an NSAP of any addressing scheme conforming to ISO/IEC 8348:1993 Addendum 2 since it satisfies the encoding and maximum length requirements of a DSP (see table H.1). In particular, the IEEE 802 MAC address constitutes a valid subaddress of the CCITT E.164 address plan since it satisfies the encoding and maximum length requirements as discussed in H.2.2. Also, for the well-known private subnetwork addressing plans (which are all hierarchical), the MAC address can be viewed as the equivalent of a subaddress augmentation to the subnetwork address, thus giving the composite address an NSAP format.

— For services involving call setup (e.g., circuit switching, CCITT X.25 packet service), the called/calling address IEs of CCITT Q.931 may be used to convey a CCITT E.164 number, any address conforming to ISO/IEC 8348:1993 Addendum 2, or through the network-specific and private network features, private network addresses. Where the address shall be carried in each PDU (e.g., CLNP, TCP/IP), the conveyance shall be effected using the protocol under consideration.

— CCITT E.164, CCITT X.121, and other ISO/IEC 8348:1993 Addendum 2 conforming addresses, unlike MAC addresses, are not "permanent" when assigned by the network administrator; they are likely to have geographical dependence and therefore it may be desirable not to tightly couple them to the ISTEs, which could either move frequently or have their MAC cards with the permanently imprinted addresses changed.

— An AU can, and in many cases is likely to, act as the IWU, and its interface to an external subnetwork can be regarded as the SNPA of the IEEE 802.9 system to the subnetwork where the subnetwork is ISDN or ISO/IEC 8348:1993 Addendum 2 conforming. Where the subnetwork is a private network external to the IEEE 802.9 subnetwork, the interface may still be viewed as the point at which the IEEE 802.9 subnetwork is identified by that network. Besides, since the AU constitutes the central point for the administration of the IEEE 802.9 system, it would be the appropriate point to interface to any external agency for provisioning.

With the above principles and the desire for simplicity and minimal modification to the ISTE, the basis of the addressing scheme for IEEE 802.9 is as follows:

a) Assign the subnetwork address to the AU to which the ISTE is connected and combine the subnetwork address with the MAC address of the ISTE to constitute the NSAP address for the ISTE. This is the network address that is used to communicate with the subnetwork for services provided by the subnetwork.

b) Where multiple AUs exist, preference is given to assigning one subnetwork address to the common point of attachment to the external subnetwork. However, the use of multiple subnetwork addresses per IEEE 802.9 subnetwork (i.e., for the AUs) and indeed per AU is not precluded. The latter will arise where an ISTE needs a subnetwork address distinct from that of the AU.

c) Where a connection setup is involved in the provision of service, the subnetwork address is conveyed from the ISTE to the AU in the CCITT Q.931 message with the ISTE MAC address as the subaddress. Where the address shall be provided in each PDU, the subnetwork protocol shall be used to convey the subnetwork address and where supported, the MAC address as the subaddress. Where the subaddress capability is not supported, each ISTE shall be assigned its own unique subnetwork address. This uses up considerably more addresses.

   The AU acts as the intermediate system between the IEEE 802.9 subnetwork and an external network and performs any address mapping required for communication with the external subnetwork.

The application of the above principles is further elaborated below for ISDN, networks conforming to ISO/IEC 8348:1993 Addendum 2, and private networks.

— For networking involving ISDN services, the CCITT E.164 address (the ISDN number) is assigned to the AU to which the ISTE is connected and the CCITT E.164 number is combined with the MAC address of the ISTE to constitute the NSAP address for the ISTE. This is the NSAP that is used in any communication involving the use of Layer 3 services. Where multiple AUs exist, preference is given to assigning one CCITT E.164 address to the common SNPA to external ISDNs, with provision for multiple CCITT E.164 addresses per IEEE 802.9 subnetwork and/or AU.

— For non-ISDN ISO/IEC 8348:1993 Addendum 2 conforming subnetworks, e.g., CCITT X.121, the subnetwork address is assigned to the AU to which the ISTE is connected and the subnetwork address is combined with the MAC address of the ISTE to constitute the NSAP of the ISTE. This is the NSAP that is used to communicate with the subnetwork for services provided by the subnetwork. Where multiple AUs exist, preference is given to assigning one subnetwork address to the common SNPA to the external subnetwork, with provision for the use of multiple subnetwork addresses.

— For subnetworks that do not conform to ISO/IEC 8348:1993 Addendum 2, e.g., private networks, the subnetwork (private network) address is assigned to the AU to which the ISTE is connected and the subnetwork address is combined with the MAC address of the ISTE to constitute the equivalent NSAP. Where multiple AUs exist, preference is given to assigning one subnetwork address to the common point at which the IEEE 802.9 subnetwork attaches to the external subnetwork, with provision for the use of multiple subnetwork addresses. Furthermore, the combined network-specific number type and private network numbering plan, and/or the user-specified subaddress feature of CCITT Q.931 are used to convey the subnetwork address as the calling/called party address and the MAC address as the subaddress.

The following subclauses examine how the above principles are applied to satisfy the requirements outlined in Chapter 1, namely,

a) Intra-AU communication
b) Inter-AU communication
c) ISTE-to-other IEEE 802 TEs
d) ISTE-to-ISDN communication
e) ISTE to other ISO/IEC 8348:1993 Addendum 2 conforming subnetworks
f) Subnetworks that do not conform to ISO/IEC 8348:1993 Addendum 2

H.3.2 Intra-AU addressing

Figure H.12 illustrates the intra-AU addressing. Intra-AU addressing encompasses both MAC services and ISDN services. For MAC services, all of the ISTEs connected to the AU are capable of recognizing MAC addresses. Therefore, any two ISTEs can exchange MPDUs by means of the MAC address; the AU provides its normal relay function. Thus, for any two ISTEs connected to the same AU, the MAC address suffices.

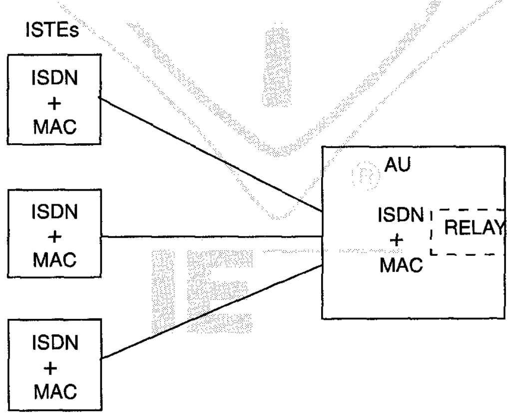


Figure H.12—Intra-AU addressing

Access to ISDN services is by means of the D channel or potentially the P_D channel. When CCITT Q.931 is invoked, the calling and called party number information shall be provided in the CCITT Q.931 call setup message. Thus, the ISTE shall identify its own ISDN address, in addition to providing the address of the other ISTE that it wishes to call. Using the principles outlined above, the ISTE will consider as its NSAP, the combination of the CCITT E.164 address of the AU to which it is attached and its MAC address. The CCITT E.164 will constitute the IDI of its NSAP and its MAC address will constitute the DSP of its NSAP.

When initiating communication for ISDN services, the ISTE will supply the CCITT E.164 address of the AU as the calling party number and its own MAC address as the calling party subaddress. The type of number field will be set to international number, since the resulting NSAP conforms to ISO/IEC 8348:1993

Addendum 2, and the numbering plan identification will be set to ISDN/telephony numbering plan. Similarly, the ISTE will supply the CCITT E.164 address of the AU as the called party number and the MAC address of the destination ISTE as the called party subaddress.

Here, since both the calling and called parties are on the AU, the calling and called party numbers are, in reality, never needed; the subaddresses would suffice. This may lead to a suggestion of an abbreviated addressing scheme using the MAC address in the calling/called party IEs. With that approach, the network-specific and private numbering plan features of the calling/called party number IEs would be used for intra-AU communication. A consequence would be an addressing format different for intra-AU communication than for communication with external ISDNs since the international number and ISDN/telephony numbering plan provision needs to be used when communicating with external ISDNs. This is discussed further in H.3.4.

H.3.3 Inter-AU addressing

Figure H.13 illustrates the case of multiple AUs. As in the intra-AU case, two modes of communication need to be considered: communication to MAC services and to ISDN services. The case for MAC services is similar to the intra-AU case except that it behaves like a network of multiple bridges and a scheme similar to the IEEE 802.1 spanning tree will need to be used. From the perspective of addressing, however, the MAC address suffices.

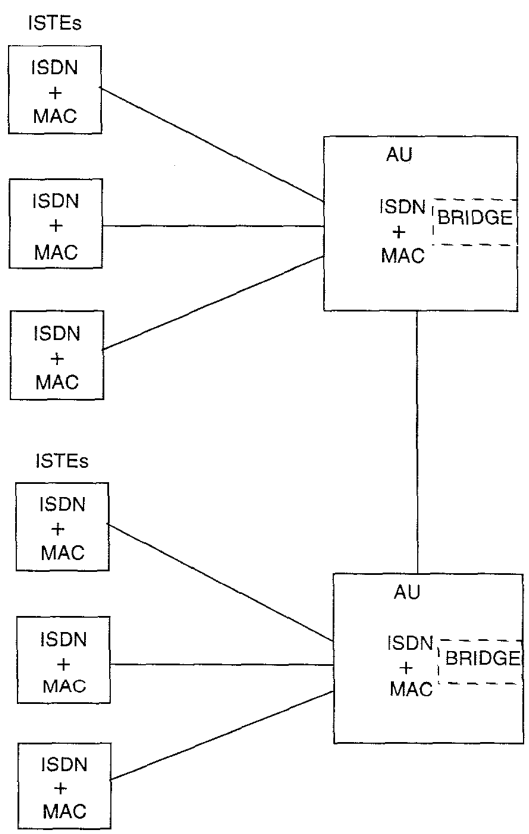


Figure H.13—Inter-AU addressing

With respect to ISDN, since the two AUs may have different CCITT E.164 addresses, the calling and called party numbers may now be different. Thus, the CCITT E.164 number of the AU to which the destination ISTE is attached is provided for the called party number IE. All other provisions are similar to the intra-AU case.

Of course, since a primary purpose of the CCITT E.164 address is to communicate with external ISDNs, it is possible to administer the IEEE 802.9 network in such a way that only one CCITT E.164 address is supplied for the whole subnetwork. This case then becomes identical to the intra-AU case.

H.3.4 Communication with other IEEE 802 TEs

Figure H.14 illustrates communication between ISTEs and other IEEE 802 TEs. Since the IEEE 802.9 ISTE shares the same MAC addressing structure and scheme as other IEEE 802 TEs, no further addressing capability is needed. The ISTEs transmit PDUs with MAC addresses; they are received by the AU, which acts as a bridge to the other IEEE 802 LAN. Because of format differences, some of the fields of the other IEEE 802 MPDUs will have to be mapped into appropriate fields in the IEEE 802.9 MPDU. In some cases, the fields may not be meaningful, and default values may have to be used. Also, the different maximum MPDU sizes may need to be reconciled. This is the same problem faced in bridging dissimilar MACs in IEEE 802 and therefore will not be discussed in further detail here.

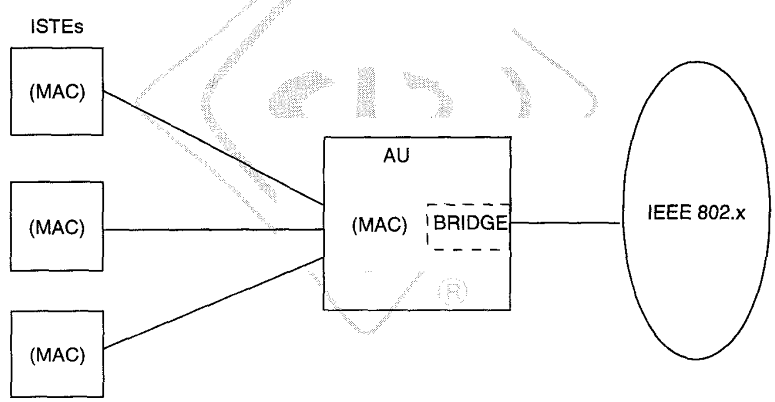


Figure H.14—Communication with other IEEE 802 TEs

H.3.5 Addressing a TE connected to an ISDN

Figure H.15 illustrates the ISTE-to-ISDN communication. Access to ISDN services is by means of the D channel or potentially the P_D channel. The calling and called addresses shall be provided in the CCITT Q.931 message. Thus, when the ISTE is calling an ISDN entity, it shall identify its own ISDN address, in addition to providing the address of the entity that it wishes to call.

As in the intra-AU case, the ISTE will consider as its NSAP, the combination of the CCITT E.164 address of the AU to which it is attached and its MAC address. The CCITT E.164 will constitute the IDI of its NSAP and its MAC address will constitute the DSP of its NSAP. When initiating communication to an ISDN, the ISTE uses CCITT Q.931 to access the AU using the CCITT E.164 address in the calling party number IE and the MAC address in the calling party subaddress IE. It will supply the AU's CCITT E.164 address as the calling party number information and its own MAC address as the calling party subaddress information. The CCITT E.164 address of the ISDN entity that the ISTE is calling will be provided in the called party number information field. Where the ISDN entity has a subaddress, it will be provided in the called party subaddress

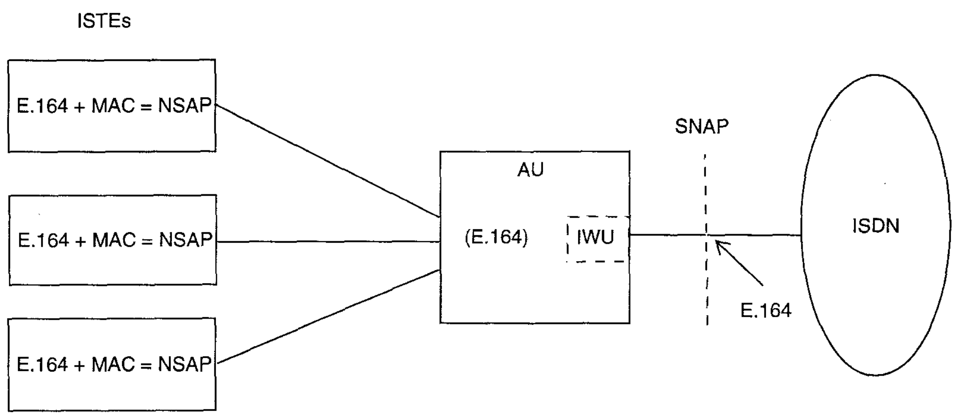


Figure H.15—Addressing between an IEEE 802.9 ISTE and an ISDN entity

information field. The type of number field will be set to international number and the numbering plan identification will be set to ISDN/telephony numbering plan.

Similarly, ISDN calls destined for a particular ISTE will have to use the subaddress feature of CCITT Q.931. The calling entity will have to supply the CCITT E.164 address of the AU as the called party number information and the MAC address of the destination ISTE as the called party subaddress information unless the ISTE is provided a CCITT E.164 address separate from the AU's. In the latter case, the subaddress feature is not required and the MAC address need not be used. Note that if the abbreviated addressing discussed in the intra-AU case were to be used, an external ISDN communicating with the IEEE 802.9 subnetwork would have to use the network-specific and private numbering plan provisions of the calling/called party number IE, which could result in translation/routing difficulties for intermediate networks between the originating ISDN and the IEEE 802.9 subnetwork. Interworking with or routing through intermediate subnetworks is greatly facilitated by the international number and ISDN/telephony numbering plan provisions.

Note that except where there is an ISTE with a different CCITT E.164 number from the AU's, the AU can, in effect, always ignore the called address IE for incoming calls except to check for incorrectly delivered calls. Likewise, the ISTEs could ignore providing the calling address information since the AU always knows it to be its own CCITT E.164 address.

H.3.6 Addressing entities on other types of ISO/IEC 8348:1993 Addendum 2 conforming subnetworks

Figure H.16 illustrates the ISTE-to-ISO/IEC 8348:1993 Addendum 2 conforming subnetwork communication. First, for the ISTE to communicate with the subnetwork, it shall be identifiable by the subnetwork through the subnetwork's addressing scheme. When the ISTE is calling an entity on the subnetwork, it shall identify its own address, in addition to providing the address of the entity that it wishes to call and vice-versa. Secondly, the calling and called addresses may be conveyed using either the CCITT Q.931 protocol or the Network Layer protocol of the subnetwork.

Using the principles outlined above, the AU to which the ISTE is attached is designated the SNPA of the IEEE 802.9 subnetwork relative to the external subnetwork and the subnetwork address is assigned to the AU. The ISTE considers as its NSAP the combination of the subnetwork address of the AU to which it is attached and its MAC address. The subnetwork address will constitute the IDI of its NSAP and its MAC address will constitute the DSP of its NSAP. Thus, it will supply the subnetwork address as the calling address and its own MAC address as the subaddress.

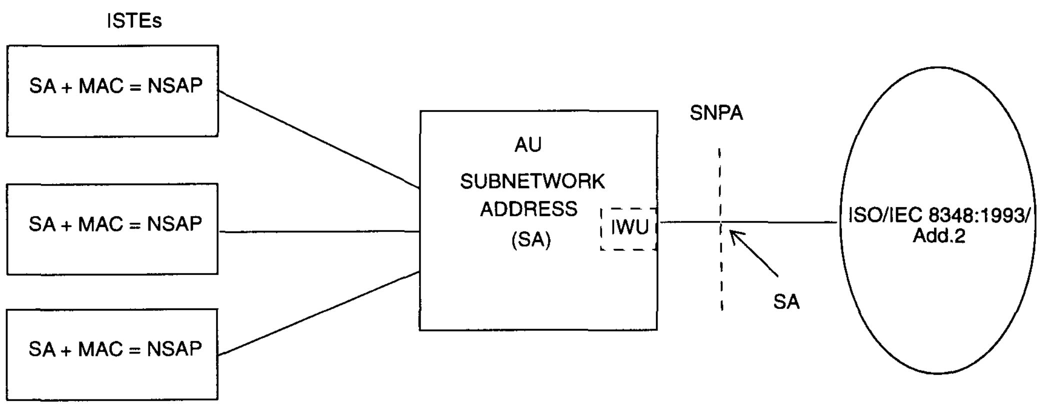


Figure H.16—Addressing between an IEEE 802.9 ISTE and an ISO/IEC 8348:1993 Addendum 2 entity

For services requiring connection setup to the subnetwork, the ISTE uses CCITT Q.931 to the AU and where the address shall be provided in the PDU, the Network Layer protocol of the subnetwork is used to convey the address to the AU. The AU, in turn, provides the address mapping to the external subnetwork.

When CCITT Q.931 is used, the subnetwork address is used in the calling party number IE and the MAC address in the calling party subaddress IE. The type of number is international number and the numbering plan identification is chosen as appropriate. For example, for CCITT X.25 subnetworks, the data numbering plan would be used. Similarly, calls destined for a particular ISTE, using CCITT Q.931, will have to use the subaddress feature and place the MAC address of the desired ISTE in the called party subaddress IE while placing the AU subaddress in the called party number IE.

Where the subnetwork protocol is used, note that since the subnetwork is ISO/IEC 8348:1993 Addendum 2 conformant, it is capable of conveying an NSAP. For originating communication from an ISTE, the SNPA is provided as the IDI of the originating NSAP and the MAC address of the ISTE is provided as the DSP of its NSAP. The destination NSAP depends on the entity being addressed on the subnetwork. It will consist of the destination SNPA and the corresponding DSP where applicable.

Note that if one subnetwork address is assigned to the IEEE 802.9 subnetwork, the AU can, in effect, always ignore the called party number information of CCITT Q.931 for incoming calls except to check for incorrectly delivered calls. Likewise, the ISTEs could ignore providing the calling party number information since the AU always knows it to be its own subaddress.

H.3.7 Communication with non-ISO/IEC 8348:1993 Addendum 2-conforming subnetworks

Figure H.17 illustrates the ISTE-to-private network communication. As for the ISO/IEC 8348:1993 Addendum 2 conforming subnetworks, for the ISTE to communicate with the subnetwork (private network), it shall be identifiable by the subnetwork through the subnetwork's addressing scheme. When the ISTE is calling an entity on the subnetwork, it shall identify its own address, in addition to providing the address of the entity that it wishes to communicate with. Again, the calling and called addresses may be conveyed with the CCITT Q.931 protocol when connection setup is involved, and with the Network Layer protocol of the private network where the address information needs to be carried with the PDU.

Using the principles outlined above, the AU to which the ISTE is attached is designated the point of attachment (equivalent SNPA) of the IEEE 802.9 subnetwork relative to the external subnetwork and the

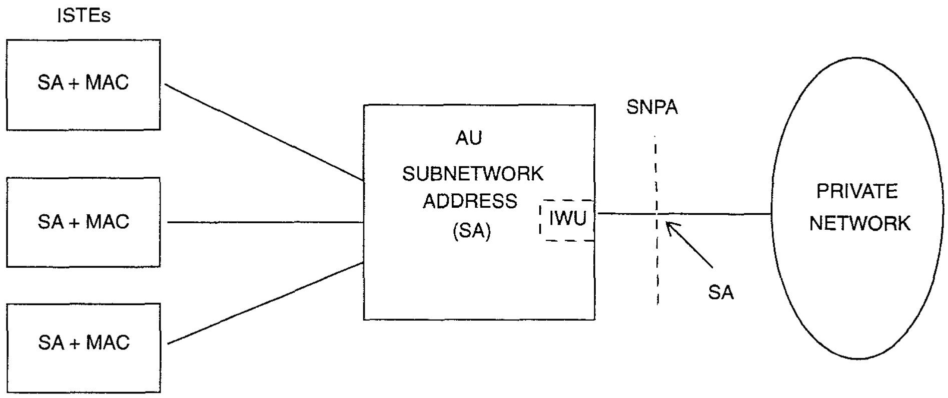


Figure H.17—Addressing between an IEEE 802.9 ISTE and a private network

subnetwork address is assigned to the AU. The ISTE considers as its full network address, the combination of the subnetwork address of the AU to which it is attached and its MAC address. Thus, it will supply the subnetwork address as the calling address and its own MAC address as the subaddress.

For connection setup from an ISTE, the subnetwork address is used in the calling party number IE of CCITT Q.931 and the MAC address of the ISTE in the calling party subaddress IE. The type of number is network-specific number and the numbering plan identification is private numbering plan. The AU provides the address mapping between the IEEE 802.9 subnetwork and the private network. Similarly, calls destined for a particular ISTE are conveyed from the AU to the ISTE using CCITT Q.931. Where the subaddress feature is supported by the private network, the MAC address would have been provided as the subaddress, which would then be placed in the called party subaddress information field. Where it is not supported, the AU shall derive the MAC address of the AU and place it in the called party subaddress information field.

With respect to using the private network's protocol for address conveyance, many such networks, such as TCP/IP, do not satisfy the conveyance requirements of ISO/IEC 8348:1993 Addendum 2. In many cases, subaddresses cannot be supported. Therefore, to use the subnetwork's Network Layer protocol for conveyance, each ISTE will have to be provided its own subnetwork address. That address is then used in the communication between the ISTE and the AU while the AU passes the ISTE's private network address to the subnetwork. For terminating communication, the external network provides the AU with the address of the destination ISTE in the PDU to be transmitted to the destination ISTE from the TE on the private network. This uses the conveyance mechanism defined by the Network Layer of the private network. The AU, in turn, passes on the PDU utilizing the addresses of the ISTE.

Annex I
Support of cell relay bearer service

(informative)

I.1 Overview

The architecture of IEEE 802.9 supports multimedia service from the desktop for presentation to a backbone local, metropolitan, or wide area network (LAN, MAN, or WAN). The technology direction of the MAN (e.g., ISO/IEC 8802-6) and WAN [e.g., broadband integrated services digital network (B-ISDN)] is to offer a cell-based service that is built upon the asynchronous transport mode (ATM) switching technology.

This annex contains an informative description of how the IEEE 802.9 architecture can be extended to support a cell bearer service.

In this annex, a "cell bearer" service is defined as a means of transferring CCITT B-ISDN conformant (or other fixed length) cells between an access unit (AU) and the integrated services terminal equipment/ terminal adaptor (ISTE/TA), within tightly controlled timing constraints. B-ISDN cells have a length of 53 octets (the first 5 octets provide the header and the next 48 octets provide user information). The transport mechanism that is described in this annex also permits the support of other sizes and structures of a "cell."

In the CCITT broadband ISDN model, all information (voice, video, and data) is conveyed using the ATM technique. This broadband architecture utilizes cells as the singular common Physical (PHY) Layer bearer which provides the transport service for packet (asynchronous), isochronous, and variable bit rate applications over a global WAN. At the originating edge of a cell-based network, B-ISDN user information is packetized by an adaption sublayer into fixed length, 53-octet units of payload (cells). These cells are then passed via cell relay (dynamic routing) at intermediate network nodes to their terminating edge of network destination, normally on high-speed (155 Mb/s) links. At the terminating edge of the cell-based network, the delivered cell is again passed to an adaption layer in which the cell header information is processed to "unpack" and return the information to its native mode form.

I.2 Cell bearer service within an IEEE 802.9 subnetwork

In IEEE 802.9, an underlying time division multiplexer (TDM) bearer is used to support several distinct isochronous bearer channels. Each channel is used either for the support of user packet (delimited bit stream oriented) service or for isochronous (continuous bit stream oriented) service. The channels within the IEEE 802.9 interface that are available for bearer service are the B1, B2, and P channels and the C channel(s) that are administered.

There are three methods by which a cell-bearer service could be accomplished within the architecture of IEEE 802.9.

- The C channel bandwidth could be allocated for this purpose.
- The cells could be wrapped and sent over an IEEE 802.9 medium access control (MAC) service within the P channel.
- A redefinition of the payload space could be provided by using the service identifier (SID) field as a discriminator for a new cell bearer service.

The methods described here do not offer an ATM cell service. Rather, they offer a fixed length cell bearer service. Consequently, ATM cells are not and do not need to be processed (interpreted) since they are simply carried over an IEEE 802.9 PHY Layer between TE/TA and AU equipment. The processing of B-ISDN or other cell header information is not within the scope of this standard.

I.2.1 Description of the available IEEE 802.9 payload bandwidth

The general format of the IEEE 802.9 TDM frame format is illustrated in figure I.1.

0	1	2	3	4	5	6	7	8 ... 63
SYN	TDM MTN	HMC	RES	D	B1	B2	AC	PAYLOAD

Figure I.1—Example of a 4.096 Mb/s TDM frame structure

With an aggregate line rate of 4.096 Mb/s, 56 of the 64 slots in the TDM frame shown in figure I.1 may be seen as "payload" space into which a B-ISDN ATM cell of 53 octets can be fitted. This provides a maximum cell user bandwidth of 3.072 Mb/s (48 × 8 × 8000 b/s) within the 4.096 Mb/s 802.9 frame period. Note that this cell user bandwidth is allocated at the expense of a total bandwidth consumption of 3.392 Mb/s (53 octets × 8 × 8000 b/s), when the 5 header octets of the cell are included. In the worst-case scenario wherein all TDM frames carry a cell, there are only 3 slots of available payload per 802.9 TDM frame (192 kb/s). This would be insufficient bandwidth to conform with the IEEE 802 requirement that there should be support for at least 1 Mb/s of bandwidth for the MAC service.

At the 802.9 aggregate line rate of 20.48 Mb/s, as illustrated in figure I.2, there is much more availability of bandwidth for a cell bearer service. In figure I.2, the format of the 20.48 Mb/s aggregate rate is illustrated. In this configuration, the first 7 octets (0 through 6) constitute the header for the 802.9 TDM frame. The access control (AC) field has been deployed in a fashion in which it is repeated for every 64 octets of the TDM payload field. This configuration results in a maximum user cell bandwidth of approximately 15 Mb/s. Note that the "extra" AC fields in slot positions 71, 135, 199, and 263 are not new channels. Rather, they are multiple instances of PhSAP=7. This AC field allocation of bandwidth leads to one row of 56 octets and 4 rows of 63 octets, into each of which a 53-octet cell can be fitted.

0	1	2	3	4	5	6
SYN	TDM MTN	HMC	RES	D	B1	B2

AC	8	63 OCTETS OF PAYLOAD SPACE	70
AC	72	63 OCTETS OF PAYLOAD SPACE	134
AC	136	63 OCTETS OF PAYLOAD SPACE	198
AC	200	63 OCTETS OF PAYLOAD SPACE	262
AC	264	56 OCTETS OF PAYLOAD SPACE	319

Figure I.2—Example of a 20.48 MB/s TDM frame structure

There may be a need to assign one or more wideband isochronous C channels, and this may lead to contention for TDM slots. Limitations of overall bandwidth in the payload field mean that individual isochronous, cell, and packet-based services will need to be engineered to ensure acceptable performance for the combination of concurrent service needs. The IEEE 802.9 dynamic bandwidth management procedures (see I.4) provide for complete flexibility.

I.2.2 Provision over a C channel

The isochronous traffic supported by IEEE 802.9 pre-allocates, on a channel exclusive basis, bandwidth that would otherwise be used for packet services. Wideband isochronous channels capable of carrying B-ISDN cells can be obtained using the dynamic bandwidth management facilities of IEEE 802.9. The bandwidth allocated to support a cell bearer service can be of any size desired by selecting an appropriate C channel bandwidth. Note, however, that such a method does not provide any "cell service." As such, it merely offers an isochronous bearer channel over which an appropriate user-defined cell delimiting scheme could convey a cell service. Furthermore, the allocated C channel bandwidth must be sufficient to support the highest rate of cell traffic that might be required and must also be commensurate with the need to ensure acceptable cell service access times. This must be achieved without unduly sacrificing the IEEE 802.9 packet service bandwidth.

I.2.3 Provision within A MAC frame carried over the P channel

It is possible to propose that the cell-based traffic be wrapped in the IEEE 802.9 MAC frame and carried across the P channel within the 802.9 subnetwork. Packet traffic, carried by the IEEE 802.9 MAC over the P channel, is typically to/from user applications that operate in a store and forward manner that will, within reason, tolerate significant frame relay transmission delays. The IEEE 802.9 MAC service access characteristics are subject to the delays inherent in the AU's multiple TE/TA request arbitration process. Since the complete specification of the AU is not within the scope of this standard, it is not possible to guarantee acceptable access for a cell bearer service over an IEEE 802.9 MAC service.

I.2.4 Provision over a new service definition within the payload field

An alternative to transport within a C channel or the MAC frame (P channel) would be to use the SID field to designate a cell-based bearer service. An advantage of this approach is that it is a more efficient method of reconciling these conflicting demands to carry cells only when they are ready for transmission, and to do so preemptively within the P channel portion of the TDM frame. This will avoid the inefficient loss of packet bandwidth that arises from the use of a dedicated isochronous channel for bursty traffic applications.

This IEEE 802.9 cell bearer service may be summarized as operating under the following rules:

a) It is restricted to the delivery of complete cells only.
b) It supports multiple cell types (these may include variable length cells).
c) It takes into account the need to support isochronous channels, and can avoid TDM slot allocation conflicts.

I.3 Cell access protocol description

In order to provide for a new bearer service definition for the IEEE 802.9 payload field, the following architectural changes would be advised:

a) The SID field would need a unique value to specify this service.

b) The interpretation of the start of frame (SOF) indicator, located in the AC field of the TDM frame, would be redefined to indicate either of the following:

1) A packet (MAC or LAPD frame currently) commences in the first P channel octet allocated in the next row (this is consistent with the present definition)

2) A preemptive cell begins in the next P channel octet in the row following the AC field containing the SOF = "1"

c) To distinguish between the start of a packet (MAC or LAPD frame) and cell service, for preemptive use of idle or packet service P channel bandwidth, the first octet in the row following the SOF = "1" would contain an appropriate SID value (this is consistent with the current use of the SID for packet service use).

d) For the case of terminal notification of bearer service capability, strong consideration should be given to the adoption of a "new" mode. For the purpose of discussion, the Mode 4 identifier would be considered in the HMC's template identification MODE field as specifying the ability to handle all of the Mode 3 functional responsibility, plus provide fixed length cell bearer support.

I.4 Packet and cell service priorities

The IEEE 802.9 PHY Layer supplies a "P channel ID" by which both packet and cell services are recognized. The packet and cell service are both carried in this P channel.

To ensure acceptable access times for the conveyance of cells, their transmission must necessarily preempt any other use of the payload channel that may already be in use by a packet service (queued for transmission). Therefore, SID encodings shall recognize those values that identify the preemptive use of the payload channel for cell service use.

The assignment of SID values to other specific services is a matter for further study. The provision of SID values for private network use is also a matter for further study.

I.5 Packet and cell service model

The support for more than one delimited frame bearer service is realized by the appropriate encoding of the SOF and SID fields. Unique values in the SID field define that the framed information following (either packet- or cell-based) is not IEEE 802.9 MAC frame data.

Architecturally the service access to the PHY Layer may be modeled as shown in figure I.3. A thin functional sublayer, the service convergence sublayer (SCS), is inserted between the PHY Layer and the MAC sublayer. In accordance with OSI conventions, the SCS sublayer shall either be modeled as part of the PHY or as part of Layer 2. This SCS function is better suited to be described as a Layer 2 function. This permits the treatment of cell and MAC framed services to be handled independently, ensuring that the MAC services are in conformance with ISO/IEC 10039:1991.

The P channel and the access control channel between the PHY and the MAC sublayer are processed by the SCS to effect a "segmentation" of the bit stream presented to the P channel. Functionally, this is implemented in the SCS by decoding the SOF and SID fields to correctly discriminate between the multiple packet and cell bearer services. In addition, the SCS shall provide the inter-service arbitration needed to implement a preemptive cell service.

The model that is illustrated in figure I.3 may be viewed as being backward compatible with the normative parts of this standard.

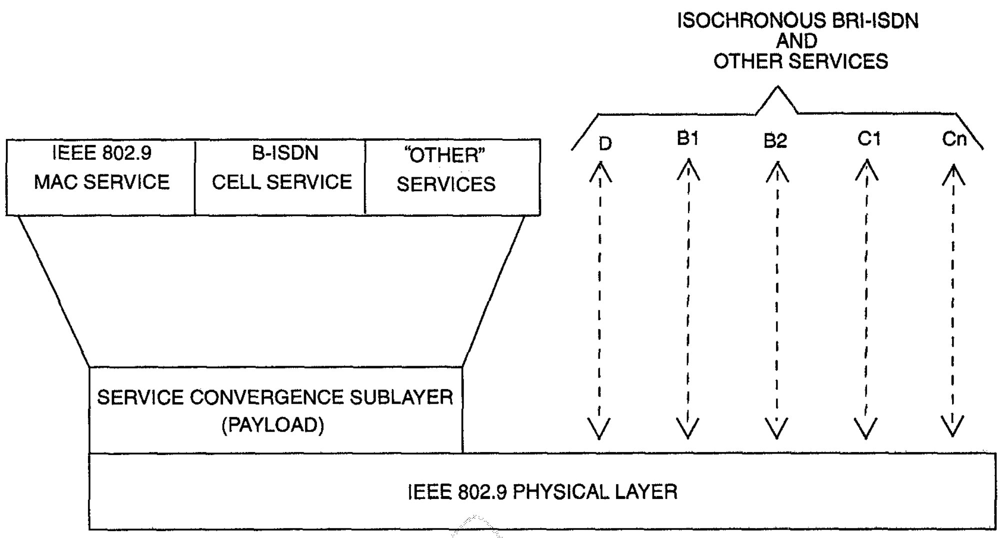


Figure I.3—Layer model for cell bearer service in IEEE 802.9

I.6 Variable length cell bearer services

The length of a cell may be specified by default, such as that the cell length will be 53 octets. Alternatively, the SID assignment may be defined such that variable length cells can be supported.

The support of a "variable length" cell service will require that a "length" field be defined to immediately follow the "variable length" SID discriminator. Upon reception of the correct number of octets per the "variable length" value, the P channel slots would be accorded to the MAC service (consistent with SOF assignment across that MAC service).

In a variable length cell service, it would not be necessary that a cell (fixed length or variable length) be conveyed in the P channel space between AC fields. With reference to figure I.3, in a multiple row P channel implementation, if a cell "crosses" an AC field boundary before its length is exhausted, the cell reception would continue until the correct "count" value is reached. Remaining P channel slots could be used for MAC frame transport until the next AC field were encountered.

I.7 Multiple cell services

A cell may be preceded by a virtual cell circuit identifier (VCCI) octet to identify multiple cell bearer channels. In this manner, support could be afforded to the multiplexing of a one-to-many cell transmission service. Thus, one TE/TA device could communicate with more than one backbone cell service. An implementation that supports the VCCI would require that the IEEE 802.9 B-ISDN bearer service use 54-octet cells. This definition can be supported with the 4.096 Mb/s aggregate rate.

With a VCCI, a B-ISDN conformant cell would become 54 octets in length. The multiplexing support of multiple cell bearer virtual channels as provided with an ATM approach would require "looking" within the cell to offer a similar service. This proposed IEEE 802.9 VCCI field would eliminate this processing step of interpreting the ATM cell.

A "length" octet could be added (the total cell size would become 55 octets) and then support of variable sized cells could be supported, subject to the limitations inherent in the segmentation of the 802.9 TDM payload space.

I.8 Inter-service contention and management

Within the HMC field's template identification MODE subfield (refer to 6.5.5.3), the identification of a capability of supporting a cell service "mode" should be defined. This would aid in the network management initialization of an IEEE 802.9 subnetwork that include nodes that may or may not support cell bearer services.

As multiple "modes" (TE class types) are added, there will be concern with how the management of P channel payload space is handled equitably. It is the end user's responsibility to correctly engineer this allocation of available bandwidth. It is not within the scope of IEEE 802.9 to define standards for the actual allocation of bandwidth. Chapter 11 and annex D provide the means by which allocation of bandwidth may be effected.

I.9 Cell service/PHY service interface

There is a need to specify the PHY service interface that is responsible for establishing the cell bearer service. This clause defines the interface "signals" necessary to support such a service.

A cell transmitter requires a "Send Cell Request" capability to alert the serving SCS sublayer that a cell is queued for transport, and a "Send Cell Enable Response" to alert the user when a cell slot has become available for transmission. A request for transmission should only be invoked when an entire cell is queued for transmission. The rate of occurrence of the AC fields will serve to guarantee a maximum delay. The incoming cell shall be delivered to the IEEE 802.9 cell service interface (see figure I.4) when the "Send Cell Enable Response" is activated.

The receiver of a cell bearer service shall always accept cells that are delivered to it. The delivery rate to the cell bearer user can never be delivered at a rate that exceeds the allocated transmission channel.

The following interface signals are proposed to be added to those listed in the IEEE 802.9 service interface description (see 7.4):

Send_Cell_Request	This SCS sublayer input indicates that a cell is awaiting transmission.
Send_Cell_Enable_Response	This SCS sublayer output indicates that available TDM slots are ready and awaiting cell transport. Note that the PHY clock and data signals associated with P channel delivery actually support this transmission.
Cell_Send_Complete	This SCS sublayer input indicates that the last octet of the current cell has been sent. This signal re-enables the use of any further P channel payload bandwidth for use by the packet services (e.g., IEEE 802.9 MAC).
Cell_Receive	This SCS output indicates that a cell is being received. This signal becomes "active" when a cell SID is detected and becomes inactive when the correct "count" of cell octets has been received.

I.10 Service priority and preemption

Cell traffic should be given priority over packet traffic. This is essential to ensure that the cell traffic is offered an acceptable quality of service (QOS).

Since there may be more than one "packet" service (MAC, "Secure" MAC, LAPD, etc.) and also more than one "cell" service (variable, fixed length), cell services should always preempt packet services. The actual order of cell service priorities is not within the scope of this standard. However, these priorities are differentiated by the use of the SID field. Similarly, the discrimination of packet service priorities could be accomplished with the use of the SID field, but the actual encoding is a matter for further study.