ISA-S72.01-1985

Approved February 3, 1986

PROWAY-LAN Industrial Data Highway

ANSI/ISA-72.01 — PROWAY-LAN Industrial Data Highway

ISBN 0-87664-896-0

ISA
67 Alexander Drive
P.O. Box 12277
Research Triangle Park, North Carolina 27709

Preface

This standard has been prepared as part of the service of ISA toward a goal of uniformity in the field of instrumentation. To be of real value, this document should not be static, but should be subject to periodic review. Toward this end, the Society welcomes all comments and criticisms, and asks that they be addressed to the Secretary, Standards and Practices Board, ISA, 67 Alexander Drive, P.O. Box 12277, Research Triangle Park, NC 27709, Telephone (919) 549-8411, e-mail: standards@isa.org.

The ISA Standards and Practices Department is aware of the growing need for attention to the metric system of units in general, and the International System of Units (SI) in particular, in the preparation of instrumentation standards. The Department is further aware of the benefits to U.S.A. users of ISA standards of incorporating suitable references to the SI (and the metric system) in their business and professional dealings with other countries. Toward this end, this Department will endeavor to introduce SI-acceptable metric units in all new and revised standards to the greatest extent possible. *The Metric Practice Guide*, which has been published by the Institute of Electrical and Electronics Engineers as ANSI/IEEE Std. 268-1982, and future revisions will be the reference guide for definitions, symbols, abbreviations, and conversion factors.

It is the policy of ISA to encourage and welcome the participation of all concerned individuals and interests in the development of ISA standards. Participation in the ISA standards-making process by an individual in no way constitutes endorsement by the employer of that individual, of ISA, or of any of the standards that ISA develops.

The information contained in the preface, footnotes, and appendices is included for information only and is not a part of the standard.

Foreword

Efforts to develop a standard for data communications to support process and industrial control began in 1976. This work was centered in Working Group 6 of Subcommittee 65C of the International Electrotechnical Committee, supported by SP72 of the Instrument Society of America and committees in Japan and Germany. This work led to development of the PROWAY A&B (token-passing bus) draft standards.

Through ANSI, members of ISA's SP72 committee continue to serve as members of the International Electrotechnical Commission (IEC) Subcommittee 65C Working Group 6 on intersubsystem computer communications. These experts have made significant contributions to the development of a series of IEC draft standards currently under consideration for worldwide acceptance. ISA-S72.01-1985 is in complete harmony with the comparable IEC document, Draft IEC Publication 955.

In 1980, Project 802 of the Institute of Electrical and Electronics Engineers began work on a general standard for Local Area Network (LAN) communications. Several members of SP72 joined the 802 committee and thus many 802 concepts were drawn from the PROWAY effort.

In January 1983, SP72 evaluated the Token-Passing Bus proposal of the IEEE 802.4 committee and concluded that it could be made suitable for industrial control. SP72 also concluded that both the general welfare and standards harmonization would be enhanced by minimizing the number of Token Bus LAN standards.

In April 1983, SP72 began a cooperative effort with the IEEE 802 committee. This effort led to development of the ISA-S72.01-1985 PROWAY-LAN Industrial Data Highway standard for industrial process control. This standard is known as PROWAY-C by the IEC.

PROWAY-LAN is compatible with (but more restrictive than) the IEEE 802.2 and 802.4 standards for general LANS. Significant enhancements were made to both the IEEE 802.2 and 802.4 standards to allow a compatible implementation of features that provide the reliability and timeliness required for industrial control.

The following is an alphabetical list of those who participated in developing the PROWAY-LAN standard. Those marked * were active members of SP72 as of the date shown on this document.

Mark Bauer
William Brown*
Anthony Capel*
Richard Caro*
Fred Casadei
Barry Cole
Steven Cooper
Robert Crowder* - Chairman
Dorel Damsker*
Stephen Dana*
Emanuel Delahostria*
Bruce Dilger
Cesar Doreza*
Jack Dorsey - Treasurer
Robert H. Douglas*
Robert Eskeridge
Gerhard Funk
Maris Graube
Kraus Grund
Robert Guzik
Mel Hagar*
Jeffrey Hale
Robert Harold*
Thomas Harrison
Wade Harsy* - Secretary
Phillip Jacobs
Dittmar Janetzky*
Kenneth Jones
Simon Korowitz*
Bob Lawler
Jim Lindgren
Laurie Lindsay
Gunther Martin
Ronald Martin
Samuel Miles
Gene Nines*
Toshio Ogawa
Thomas Phinney
Dennis Quy*
John Ricketson
Fred Schwierske
Paul Senechal
Harvey Shepard*
Don Smith*
Mark Steiglitz
Gary Stephens
Ken Still
Kathleen Sturgis*
David Sweeton*
Allen Tanzman
Seppo Turunen
Michael Ullman
Jay Warrior*
Earl Whitaker*
Chris Wilmering
Graeme Wood
Prentiss Yates*

Special thanks are due to Anthony C. Capel, Robert S. Crowder, Dittmar Janetzky, Gunther Martin, Thomas L. Phinney, and David C. Sweeton for technical contributions and to David E. Carlson, Robert H. Douglas, Robert S. Crowder and Eugene Nines for coordinating the efforts of the IEEE 802.2 and 802.4, ISA SP72 and IEC/SC65C/WG6 committees during the joint development of this standard for industrial applications of Token Bus Local Area Networks.

This standard was approved for publication by the ISA Standards and Practices Board in June 1985.

N. Conger, Chairman
P. V. Bhat
W. Calder III
R. S. Crowder
B. Feikle
H. S. Hopkins
J. L. Howard
R. T. Jones
R. Keller
0. P. Lovett, Jr.
E. C. Magison
A. P. McCauley
J. W. Mock
E. M. Nesvig
R. Prescott
D.E. Rapley
C. W. Reimann
J. Rennie
W. C. Weidman
K. Whitman
*P. Bliss
*B. A. Christensen
*L. N. Combs
*R. L. Galley
*T. J. Harrison
*R. G. Marvin
*W. B. Miller
*G. Platt
*J. R. Williams

*Director Emeritus

Contents

Standard

PROWAY-LAN
Industrial Data Highway

I. Introduction and overview

I.1 Scope. This standard specifies those elements which are required for compatible interconnection of stations by way of a Local Area Network (LAN) using the Token-Bus access method in an industrial environment. These elements include

1) the electrical and physical characteristics of the transmission medium

2) the electrical transmission signaling method used by the Physical protocol

3) the services and signals provided at interface between the Physical and Media Access Control entities

4) the frame formats transmitted

5) the Token Bus Media Access protocol

6) the services provided at the conceptual interface between the Medium Access Control (MAC) sublayer and the PROWAY Link Control (PLC) sublayer above it

7) the PROWAY Link Control (PLC) protocol

8) the services provided at the conceptual interface between the User of PROWAY and the PROWAY Link Control (PLC) sublayer

9) management of PROWAY stations

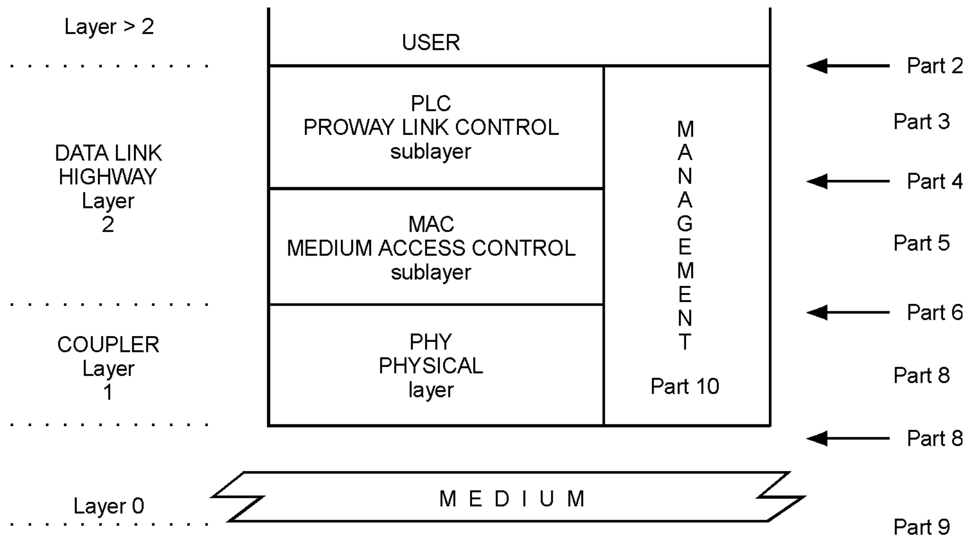


Figure I-1: LAN model

I.2 Introduction

I.2.1 Layering. Within this standard the operation of a station is specified in terms of the layered model shown in Figure I-1. The figure also shows which parts of the standard specify interfaces between layers and which parts specify the operation of the layers themselves. This figure is repeated in each part of the standard to indicate the relationship of that part to the remaining parts of the standard and to the LAN model.

I.2.2 Precedence. SParts I and 1 are introductory and explanatory in nature. Parts I and 1 are not intended to convey precise specifications. In case of a conflict between Parts I and 1 and any other part of the standard, the specifications of the other part shall take precedence.

I.2.3 Definitions. Terms that are used in a specific or specialized manner in any part of this standard are defined at the beginning of that part. General definitions are given in References 1 and 2, Appendix I-A.

I.2.4 References. Documents referenced by this standard are listed in Appendix I-A

I.3 Compliance

I.3.1 Classes of stations. This standard defines three classes of stations:

INITIATOR stations perform all functions of the Local state machine and the reception functions of the Remote state machine.

RESPONDER stations perform all the functions of the Remote state machine and none of the functions of the Local state machine. A simple device (such as an instrument) can be connected to the Local Area Network as a **RESPONDER**.

INITIATOR/RESPONDER stations perform all functions of both the Local and Remote state machines.

I.3.2 Compliance requirements. Implementations may claim compliance with this standard as one of the classes of stations given in I.3.1. Implementations that claim compliance with this station shall

1) offer the mandatory PLC-user interface and services specified in this standard for that class of station
2) support the PROWAY Link Control protocol specified in this standard for that class of station
3) support the mandatory PLC-MAC interface and services specified in this standard for that class of station
4) offer the mandatory station management services specified in this standard for that class of station
5) support the mandatory features of the Token Bus Medium Access protocol specified in this standard for that class of station
6) support the MAC-Physical interface services specified in this standard

It is strongly recommended that **INITIATOR** stations support reporting of the list of stations participating in the logical token ring.

A number of options are specified in this standard. An implementation must indicate which, if any, of these options are supported and the supported values of each station attribute.

I.4 PROWAY organization

The PROWAY Data Highway has three primary functional layers or entities:

- **PROWAY Link Control - PLC**
- **Media Access Control - MAC**
- **Physical Signaling - PHY**

Each of these layers is briefly introduced in the following sections.

The PLC and MAC sublayers together comprise the Data Link (Highway) level of the ISO Model. The PHY level comprises the Physical level of the ISO Model.

I.5 Overview of the PROWAY Link Control (PLC) layer

I.5.1 PROWAY Link Control services. PROWAY provides four basic services to its users:

- **Sending data using a confirmed** (immediate response) **protocol** from one local (originating) station to one remote (responding) station. This service is known as **Send Data with Acknowledge or SDA**.
- **Sending data without acknowledge or retry** from one local station to one, some, or all remote (receiving) stations. This less secure service is known as **Send Data with No Acknowledge or SDN**.
- One local station **requesting previously submitted information** from one remote station using the confirmed (immediate response) protocol. This service is known as **Request Data with Reply or RDR**.

I.5.2 PROWAY link control sublayer organization. The PLC sublayer functions are logically divided into two independent state machines:

1) Local state machine

2) Remote state machine

I.5.2.1 Local state machine functions. The Local state machine handles all requests from and confirmations to the local PLC user. These local requests result in transmission of request frames. Functions of the local PLC state machine include:

- accepting local user requests (excluding update_requests)
- generating request frames
- receiving response frames
- passing confirmations to the local PLC user (excluding update_confirmations)

I.5.2.2 Remote state machine functions. The Remote state machine passes indications to the remote PLC user, manages the shared data areas, and returns requested data to the local machine. Functions of the remote PLC state machine include:

- receiving request frames
- passing indications to the remote PLC user
- generating response frames
- accepting remote user requests to update a shared_data_area
- passing update confirmations to the remote PLC user

I.6 Overview of the token bus access method

I.6.1 The essence of the token bus access method

1) A token (or baton) controls the right of access to the physical medium; the station which holds (possesses) the token has control over the medium.

2) The token is passed by stations residing on the medium. As the token is passed from station to station a logical ring is formed.

3) Steady state operation consists of a data transfer phase and a token transfer phase.

4) Ring maintenance functions within the stations provide for ring initialization, lost token recovery, new station addition to logical ring, and general housekeeping of the logical ring. The ring maintenance functions are replicated among all the token-using stations on the network.

Shared media generally can be categorized into two major types. These types are broadcast and sequential. This standard deals exclusively with the broadcast type. On a broadcast medium, every station may receive all signals transmitted. Media of the broadcast type are usually configured as a Physical bus.

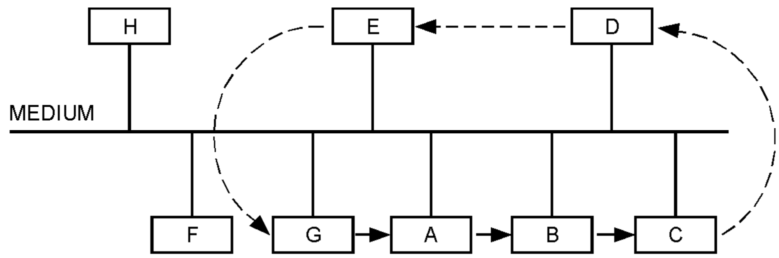


Figure I-2: Logical ring on Physical bus

In Figure I-2, note that the token bus access method is always sequential in a logical sense. That is, during normal steady state operation, the right to access the medium passes from station to station. Further, note that the physical connectivity has little impact on the order of the logical ring and that stations can respond to a query from the token holder even without being part of the logical ring. (For example, stations H and F can receive frames but cannot initiate a transmission since they will never be sent the token.)

The Medium Access Control (MAC) sublayer provides sequential access to the shared bus medium by passing control of the medium from station to station in a logically circular fashion. The MAC sublayer determines when the station has the right to access the shared medium by recognizing and accepting the token from the predecessor station, and it determines when the token must be passed to the successor station.

I.6.2 General MAC sublayer functions

1) lost token timer
2) distributed initialization
3) token holding timer (for multiple classes of service and ring maintenance)
4) token rotation timer (for multiple classes of service and ring maintenance)
5) limited data buffering
6) node address recognition
7) frame encapsulation (including token preparation)
8) Frame Check Sequence (FCS) generation and checking
9) valid token recognition
10) new ring member addition
11) node failure error recovery
12) allowing an immediate PLC response to or acknowledgment of a frame sent by the token holder
13) retry transmission for certain classes of service.

I.7 MAC layer internal structure. The MAC layer performs several functions which are loosely coupled. The descriptions and specifications of the MAC sublayer in this standard are organized in terms of one of several possible partitionings of these functions. The partitioning used here is illustrated in Figure I-3, which shows four asynchronous logical "machines," each of which handles some of the MAC functions, as discussed below.

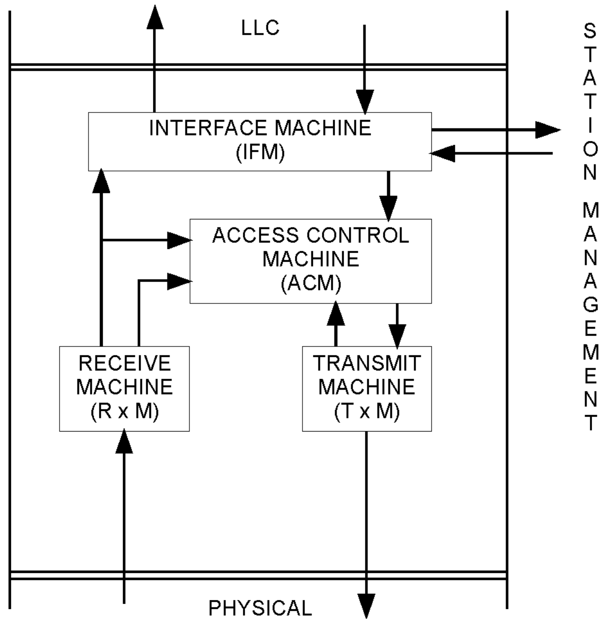


Figure I - 3: MAC layer functional partitioning

I.7.1 Interface Machine (IFM). This machine acts as an interface and buffer between the PLC and MAC sublayers and between Station Management and the MAC sublayer. It interprets all incoming service primitives and generates appropriate outgoing service primitives. This machine handles the mapping of "quality of service" parameters from the PLC view to the MAC view, and generation of a response frame when required. The IFM handles queuing of service requests sent to a PLC protocol data unit. It performs the "address recognition" function on received PLC frames, accepting only those addressed to this station. Finally, it generates an immediate response when requested by the initiating station.

I.7.2 Access Control Machine (ACM). This machine cooperates with the ACMs of all other stations to control transmission on the shared bus medium. The ACM manages multiple MAC access classes in order to provide different levels of "quality of service" to the PLC layer. When required, it will wait for a response or acknowledgment of a transmitted frame from the remote station. If the required response is not received, the local ACM will retry the transmission. The ACM is also responsible for initialization and maintenance of the logical ring, including the admission of new stations. Finally, it has responsibility for the detection of and, where possible, recovery from faults and failures in the token bus network and for informing station management of changes in the membership of the token ring. The ACM is not required in a **RESPONDER** station.

I.7.3 Receive Machine (RxM). This machine accepts atomic symbol inputs from the physical layer, assembles them into frames, which it validates and passes the frames to the IFM and ACM. The RxM accomplishes this by recognizing the frame Start Delimiter (SD) and the frame End Delimiter (ED), checking the Frame Check Sequence (FCS) and validating the frame's structure. The RxM also identifies and indicates the reception of **noise_bursts** and the **bus_quiet** condition.

I.7.4 Transmit Machine (TxM). This machine generally accepts a data frame from the ACM and transmits it, as a sequence of atomic symbols in the proper format, to the physical layer. The TxM builds a MAC protocol data unit by prefacing each frame with the required preamble and SD, and appending the FCS and ED.

I.8 Token bus access method characteristics

An understanding of the basic characteristics of the token bus access method assists in understanding where and when a token-passing bus is an appropriate LAN technology.

Some of the important features of the token bus access method are as follows:

1) The method is efficient in the sense that under high offered load the coordination of the stations requires only a small percentage of the media's capacity.

2) The method is fair in the sense that it offers each station an equal share of the media's capacity. It does not, however, require any station to use its full share.

3) The method permits multiple priorities.

4) The method supports open loop (no response) and closed loop (with response or acknowledgment) data transfers.

5) The method coordinates station transmissions to minimize and control interference with other stations.

6) The method imposes no additional requirements on the media and the modem capabilities over those necessary for transmission and reception of multi-bit, multi-frame sequences at the specified bit error rate.

7) In the absence of system noise, the method provides computable, deterministic, worst case bounds on access delay, for the highest priority class of service, for any given network and loading configuration.

8) Periods of controlled interference are distinguishable; system noise measurements are possible during the remaining periods.

9) The method places minimal constraints on how a station which momentarily controls the media may use its share of the media's capacity. In particular, the access method allows a station to use request/reply access methods during that station's access periods.

10) The method permits the presence of a large number of low-cost reduced-function **(RESPONDER)** stations in the network with one or more full-function **(INITIATOR)** stations. (It is assumed that at least one **INITIATOR** station is needed to make the system operational, for example, to initialize.) A **RESPONDER** station does not require Media Access Control logic.

11) Use of the immediate response feature in conjunction with the confirmed (SDA or RDR) data transfer services allows protection against loss or duplication of frames at any desired level of assurance.

I.9 Overview of the Physical layer and media

I.9.1 Summary of phase continuous FSK (Frequency Shift Keying) Physical layer

Transmit level: +64 to +66 dB (1 mV, 75 ohm) (dBmV); i.e., approximately 2 V rms.

Receiver sensitivity: +4 to +50 dB (1 mV, 75 ohm) (dBmV), and the ability to receive one's own transmissions.

Noise floor: $\leq$ -15 dB (1 mV, 75 ohm) over a 3-7 MHz band.

Data rate: 1 Mbits/sec.

Signaling: Manchester encoding of **data**, **non_data**, and **pad_idle** symbols. The transmission symbol representations are

{HL} = **zero** }

{LH} = **one** } **data**

{LL}{HH} = pairs of **non_data** symbols containing no frequency transitions for one full MAC symbol period

Octets of consecutive **ones** and **zeros** = **pad_idle** (preamble)

Modulation: Phase continuous FSK (Frequency Shift Keying) (a form of frequency modulation).

1) Higher frequency = 6.25 $\pm$ 0.08 MHz = [H]

2) Lower frequency = 3.75 $\pm$ 0.08 MHz = [L]

Clock recovery: From transitions generated by the Manchester encoding.

I.9.2 Summary of single-channel coaxial cable bus medium

Topology: Omnidirectional bus

Cable: 75 ohm coaxial trunk cable, such as types RG-6 and semi-rigid, with flexible drop cables such as RG-59.

Recommended cable configuration: Semi-rigid trunk and flexible drop cable.

Trunk connection (TAP): 75 ohm nondirectional passive impedance-matching tap.

Repeaters: Active repeaters and amplifiers are not used in a PROWAY system.

I.9.3 Alternate physical and media implementation. Additional physical layers, data rates, and compatible media (e.g., 5 Mbits/sec, phase coherent or fiber optics) as defined in Chapters 12/13 and 16/17 of Reference 4 are compatible implementations of this standard.

I.10 Standard organization. This standard is organized in 10 parts, which are summarized below.

Part I serves as an introduction. It begins with a general discussion of the PROWAY link control services and token-passing bus access methods. The PLC and MAC sublayer functional partitioning used in subsequent parts is introduced here. Features of the token-passing bus access method are next reviewed. The Physical layer and Media are then surveyed.

Appendix I-A lists standards referenced by this standard.

Part 1 gives the functional requirements for industrial local area networks.

Part 2 outlines the user interface to the Process Data Highway (PROWAY).

Part 2A details the user interface for transmission of data through PROWAY; i.e., it details the user interface to the PROWAY Link Control (PLC) sublayer.

Part 2B defines the user interface to station management functions.

Appendix 2-A provides an overview of the operation of the Data Link layer (PLC and MAC sublayers combined) and the user interactions with PROWAY.

Part 3 specifies the PROWAY Link Control (PLC) sublayer.

Part 3A gives definitions that are applicable to the PLC state machines and specifies mandatory requirements of the PLC sublayer.

Part 3B defines the functionality of the PROWAY Link Control (PLC) sublayer by means of state tables.

Part 3C details the format of Protocol Data Units (L_pdu) exchanged between cooperating PLC entities.

Part 4 details the interfaces between PLC and MAC sublayers. It defines the services and interfaces provided to the PLC sublayer.

Part 5 specifies the Medium Access Control (MAC) sublayer and details the format of frames exchanged between cooperating MAC entities.

Part 5A discusses the basic concepts of the medium access protocol and provides an informal description of the actions in each state of the Access Control Machine.

Part 5B contains definitions of essential MAC terms and components and specifies those requirements for the MAC sublayer protocols which are not covered elsewhere. It specifies the Interface machine and references the Receive and Transmit machines in Reference 4, Appendix I-A.

Part 5C specifies the MAC Access Control Machine in Reference 4. This is the definitive specification of the token-passing bus-MAC cooperation.

It also describes the MAC layer variables, functions and procedures used in the state machine.

Part 5D details the MAC frame structure including delimiters, addressing, and the FCS. All of the frame formats which MAC handles, including MAC control frames, are enumerated.

Appendix 5-A gives an example configuration for a PROWAY network that meets the requirements of Part 1.

Appendix 5-B gives examples of 16 bit station addresses.

Part 6 details the interface between the MAC sublayer and the Physical layer.

Part 6A specifies the MAC-Physical interface services in an abstract manner and provides descriptions of the interface symbols, requests, and indications.

Part 6B defines the MAC-Physical interface implementation required when the MAC and Physical entities are separate pieces of equipment.

Part 7 is reserved for future use.

Part 8 specifies the Phase-Continuous-FSK Physical layer entity and its interface to the medium.

Part 9 gives requirements for a single-channel coaxial cable bus medium.

Appendix 9-A gives guidance for configuring and installing single-channel coaxial cable bus networks.

Part 10 defines PROWAY station management.

Part 10A informally defines the activities of the station management entity with respect to local requests by means of tables of required actions.

Part 10B defines the interface between the station management entity and the PLC sublayer.

Part 10C details the interface between the station management entity and the MAC sublayer.

Part 10D details the interface between the station management entity and the Physical layer.

Part 10E specifies maintenance of the list of stations participating in the token ring (the active station list) and the format of Protocol Data Units (Mgt_pdu's) exchanged between cooperating management entities in maintaining this list.

Part 10F specifies a communications reset capability.

Appendix 10-A gives examples of active station list activity.

Appendix 10-B provides recommendations for preconditions to entering and leaving the logical token ring.

I.11 Correspondence between the PROWAY and IEEE 802 standards. The parts of the PROWAY-LAN standard are related to the following sections of IEEE 802 standard.

PROWAY-LAN Part	IEEE Standard and section
PROWAY-Lan Part I	802.4—Section 1
PROWAY-Lan Part 2A	802.2—Section 2
PROWAY-Lan Part 2B	______
PROWAY-Lan Part 3A	802.2—Definitions
PROWAY-Lan Part 3B	802.2—Section 6 and Single-Frame
PROWAY-Lan Part 3C	802.2—Section 3,5
PROWAY-Lan Part 4	802.4—Section 2
PROWAY-Lan Part 5A	802.4—Section 5
PROWAY-Lan Part 5B	802.4—Section 6
PROWAY-Lan Part 5C	802.4—Section 7
PROWAY-Lan Part 5D	802.4—Section 4
PROWAY-Lan Part 6A	802.4—Section 8
PROWAY-Lan Part 6B	______
PROWAY-Lan Part 8	802.4—Section 10
PROWAY-Lan Part 9	802.4—Section 13
PROWAY-LAN Appendix 9-A	802.4—Appendix 13A
PROWAY-Lan Part 10A	______
PROWAY-Lan Part 10B	802.2—Management Proposal
PROWAY-Lan Part 10C	802.4—Section 3
PROWAY-Lan Part 10D	802.4—Section 9
PROWAY-Lan Part 10E	______

I.12 Compatibility with IEEE 802. This standard defines the features which are mandatory for conformance to PROWAY specifications. Conformance to IEEE 802 specifications requires additional features as defined in the IEEE 802.1, 802.2, and 802.4 standards (see References 2-4 of Appendix I-A).

Implementation specifics for Chapters 2, 3, 6, and 10 are subject to harmonization with the future work of the IEEE 802 committee.

In any case, where there is a direct conflict between this standard and a requirement of an equivalent part of one of the above IEEE 802 standards, the requirement of the IEEE 802 standard shall take precedence. Where this standard imposes additional requirements beyond those of the above IEEE 802 standards, the additional PROWAY requirements shall take precedence.

Appendix I-A

References

Ref. No.		Section No.
1)	ISO 7498-1984. Information Processing Systems Interconnection - Basic Reference Model.	All
2)	IEEE Std 802.1, Local Area Network Standard - Overview, Interworking, and Management. (Draft standard, publication pending)	I,10
3)	IEEE Std 802.2 (1985), Logical Link Control. IEEE 802.2 Proposed Draft Addendum — Acknowledged Connectionless Service, Draft 13, Oct. 1986.	I,2,3
4)	IEEE Std 802.4 (1987), Token-Passing Bus Access Method and Physical Layer Specifications.	I,4,5
5)	Department of Defense, MIL Standard 1851A (1983), ADA Programming Language.	I
6)	ADA™ Programming Language.	I,5
7)	IEC Publication 79-10 (1972) Part 10: Classification of Hazardous Areas.	1
8)	IEC Publication 79-1 (1971) Part 1: Construction and Test of Flameproof Enclosures of Electrical Apparatus.	1
9)	IEC Publication 79-3 (1972) Part 3: Spark Test Apparatus for Intrinsically Safe Circuits.	1
10)	IEC Publication 79-11 (1984) Part 11: Construction and Test of Intrinsically Safe and Associated Apparatus.	1
11)	ISO 4902:1980, Data Communication — 37-Pin and 9-Pin DTE/DCE Interface Connectors and Pin Assignments.	6
12)	CCITT Yellow Book, Vol. 11 (1980), Characteristics for Balanced Double Current for General Use with Integrated Circuit Equipment in the Field of Data Communications.	6
13)	IEC Publication 348 (1978), Safety Requirements for Electronic Measuring Apparatus.	6
14)	FCC Docket 20780-1980 (Part 15) Technical Standards for Computing Equipment. Reconsidered First Report and Order, April 1980.	8
15)	NFPA National Electric Code, Article 250: Grounding; Article 800: Communication Circuits; Article 820: Community Antenna Television and Radio Distribution Systems.	8,9
16)	IEC 716-1983, Expression of the Properties of Signal Generators.	

17)	UL 94, Test for the Flammability of Plastic Materials for Parts in Devices and Appliances (Rated under V-0).	8,9
18)	UL 114, Office Appliances and Business Equipment.	8,9
19)	UL 478, Electronic Data Processing Units and Systems.	8,9
20)	CSA Standard C22.2 No. 154-M 1983, Data Processing Equipment.	8,9
21)	IEC 435 (1983), Safety of Data Processing Equipment.	8,9
22)	CCITT Yellow Book, Vol. 8, Fascicle VIII.1, Data Communications over the Telephone Network.	8
23)	CCITT Yellow Book, Vol. 8, Fascicle VIII.3, Data Communication Networks.	8
24)	CCITT Yellow Book, Vol. 8, Fascicle VIII.2, Data Communication Networks .	8
25)	CCITT Yellow Book, Vol. 3, Fascicle III.4, Line Transmission of Non-Telephone Signals.	9
26)	CCITT Yellow Book, Vol. 3, Fascicle III.4, Impedance Matching between Repeaters and Coaxial Pairs in Television Transmission.	9
27)	CCITT Yellow Book, Vol. 3, Fascicle 111.4, Annex A to Recommendation J.73.	9
28)	General Motors Unified Communications Systems Task Force of MCC/CMC Computers in Manufacturing Subcommittee, 1978, General Broadband Coaxial Cable Networks for Digital, Video, and Audio Transmission.	9
29)	Rheinfelder, W. *CATV Circuit Engineering*, TAB Books, Blue Ridge Summit, PA.	9
30)	Rheinfelder, W. *CATV System Engineering*, TAB Books, Blue Ridge Summit, PA.	9
31)	RCA/Cablevision Systems: *Design and Construction of CATV Systems*, Van Nuys, CA.	9
32)	*Basic CATV Concepts*, Theta Com CATV/TEXCAN, Phoenix, AZ.	9
33)	EIA CB8-1981, Components Bulletin, List of Approved Agencies, US and Other Countries, Impacting Electronic Components and Equipment .	8
34)	IEC Publication 255-4 (1976), Single Input Energizing Quantity Measuring Relays with Dependent Specified Time.	I
35)	US Military Standard — MIL C 39012.	8,9

1 Functional requirements

1.1 Overview. The PROWAY standard defines the protocols, interfaces, and media for layers 1 and 2 of the ISO reference model. Compliance with this standard, and with complementary standards at the higher layers will allow unambiguous communication between the devices that comprise a Distributed Industrial or Process Control System over a shared Process Data Highway (PROWAY). Compliance with these standards will enable devices from different manufacturers to cooperate in the same control system. The PROWAY-LAN standard is applicable to control systems for both continuous and discrete processes and to a wide range of factory automation systems. Industrial control systems are distinguished from other on-line, real-time computer networks in that control systems' outputs cause material or energy to move.

This standard applies to serial transmission over a single shared electrical transmission line; i.e., a coaxial cable. However, future revisions may define alternative transmission media, such as fiber optics.

The Functional Requirements given in Part I were developed prior to the other parts of the standard and were the primary basis for evaluating the technical merits of proposals for use in Parts 2 through 10 of this standard. If there are conflicts between Part 1 and other Parts of this standard, the requirements of the other Part take precedence.

1.2 Application environment and main features.

1.2.1 Application characteristics. The characteristics of the data highway should be such that they provide optimum features for use in industrial control systems and shall be applicable to both continuous and discrete processes. An industrial data highway is characterized by the following:

1) event-driven communication which allows real-time response to events

2) very high availability

3) very high data integrity

4) proper operation in the presence of electromagnetic interference and differences in earth potentials

5) dedicated intra-plant transmission lines

1.2.2 Economic versus technical factors. To achieve broad applicability it is essential that industrial data highways should be economically viable in control systems under the following conditions:

1) with low or high information transfer requirements

2) within a control room and/or while exposed to the plant environment

3) in geographically small or large plants

The economic and technical factors may need to be reconciled to achieve a balance of transmission line length versus data signaling rate.

1.2.3 Main features of PROWAY

Number of stations	**$\leq$ 100**
Length of highway	**$\leq$ 2000m**
Data signaling rate	**$\geq 1 \times 10^6$ bits/sec**
Data circuit bit error rate	**$< 1 \times 10^{-8}$**
Residual error rate	**$< 3 \times 10^{-15}$ at a bit error of $1*10^{-6}$**
Maximum user data in frame	**$\leq$ 1000 octets**
Information transfer rate	**$\geq 3 \times 10^5$ bits/sec**
High priority media access time	**$\leq$ 10 millisec**

1.3 Device types

1.3.1 Communications between control devices. Communications shall be provided among commonly used devices in process or industrial control applications.

It is intended that this standard will provide optimum performance when intelligent control devices are communicating.

1.3.2 Communications with other devices. An industrial data highway is not intended to provide an optimized interface for high-speed computer memories or peripherals. However, no devices or types of devices are excluded from exchanging data over an industrial data highway, provided they conform to the requirements of this standard.

1.3.3 Classes of stations. The data highway shall support full function **(INITIATOR)** stations and reduced function **(RESPONDER)** stations. Simple devices may be directly connected to the data highway as **RESPONDERS**.

1.4 System structure

1.4.1 Control system structure. An industrial data highway shall be capable of supporting control systems with centralized intelligence, distributed intelligence, hierarchical intelligence and combinations thereof.

1.4.2 Data highway mode of operation. The data highway shall be capable of supporting transmission of event oriented data in real time. The normal mode of operation uses transaction message pairs, such that each request message is followed by its related response or acknowledgment message; i.e., by an immediate response.

Any two stations on a single data highway shall perform direct data interchange without involving store and forward at a third station.

1.4.3 Configuration changes. The data highway shall support reconfiguration of the control system while the process is operating. During reconfiguration, a transient disturbance to the exchange of frames is permitted, provided that the data highway is able to detect such disturbance and that it can recover full operation within a time appropriate to the application.

Examples of these configuration changes are as follows:

1) extending, shortening or rerouting transmission lines

2) connecting or disconnecting stations from the transmission line

1.5 Maintenance and service features

1.5.1 Testing and fault diagnosis. An industrial data highway must include means for carrying out test and fault diagnoses on line.

1.5.2 Effect of state transitions. Any station shall be able to perform transitions from one state to another without generating bit errors between other stations. Examples of such transitions are as follows:

1) On-line/off-line

2) Power-on/power-off

3) Ready/not ready

4) Busy/not busy

5) Local/remote

1.6 Safety

1.6.1 Electrical faults. All devices used in the data highway shall be capable of withstanding the application of an allowable fault potential appropriate to the application. Application of this potential to the device's connection to the transmission line shall not damage the device nor cause it either to damage other devices, or become hazardous to personnel. Three classes of installation can be identified. They are as follows:

1) The fault potential is the power mains voltage in the area traversed by the transmission line(s).

2) The fault potential is represented by a pulse typically 2.5 kV peak with a 1 microsec rise time and 50 microsec decay time, the test procedure is given in Reference 34, Appendix I-A.

3) The fault potential is that which is generated by a lightning strike to an arbitrary point near the transmission line(s). Such a fault is typically characterized by a 10 microsec rise time to 5,000 amps and a 20 microsec fall time to half that value. This is referred to as a10/20 microsec pulse.

1.6.2 Intrinsic safety. The design of the data highway shall include consideration of possible extensions to use the equipment, or sections of it, in hazardous atmospheres.

The supplier shall state which of the four categories below his equipment will meet:

1) not suitable for hazardous atmospheres (See Reference 7, Appendix I-A)

2) flameproof construction (See Reference 8, Appendix I-A)

3) intrinsically safe (See References 9 and 10, Appendix I-A)

4) claimed to meet the requirements for intrinsic safety certification but not actually certified.

For categories (2) and (3), the supplier shall give the name of the approving authority, the class of certification and the approval certificate number.

1.7 Performance in industrial environment

1.7.1 Industrial environment

1.7.1.1 Induced noise. The noise floor on a main trunk cable installed in a typical industrial environment in accordance with Part 9 may be as high as 0 dBmV measured over a 3 to 7 MHz bandwidth.

1.7.1.2 Electromagnetic environment. The industrial environment may include an ambient plane wave field of

1) 2 volts/meter from 10 kHz through 30 MHz

2) 5 volts/meter from 30 MHz through 1 GHz

1.7.1.3 Differences in earth potential. Typical differences in earth potential for an industrial environment are as follows:

1) When the transmission medium is entirely contained in a protected area, this difference in earth potential is typically less than 10 V peak-to-peak at frequencies less than 400 Hz.

2) When the transmission medium is exposed to the plant environment, this difference in earth potential is typically less than 50 V peak-to-peak at frequencies less than 400 Hz.

3) When the transmission medium is exposed to a severe plant environment (for example a power station), this difference in earth potential may typically rise to 1000 V peak-to-peak at frequencies less than 10 MHz.

1.7.2 Data circuit bit error rate. The data highway, when installed in a typical industrial environment and in accordance with Part 9, shall exhibit a data circuit bit error rate of no more than 1×10^{-8}. The manufacturer shall provide a graph relating data circuit bit error rate to the noise floor, differences in earth potential, data rate, and other relevant factors.

1.7.3 Residual error rate. The data highway shall achieve a residual error rate of no more than 3×10^{-15} when the data circuit bit error rate is 1×10^{-6}. The manufacturer shall provide a graph which relates the residual error rate to the data circuit bit error rate.

NOTE: The corresponding rate of undetected frame errors is one error per 1000 years of data highway operation, assuming 100% utilization of a data signaling rate of 1×10^{6} bits/sec.

1.7.4 Information transfer rate. The data highway shall achieve an information transfer rate of at least 3×10^{5} bit/sec, when data circuit bit error rate is 1×10^{-6} and the data rate is 1×10^{6} bits/sec. The manufacturer shall provide a graph which relates information transfer rate to the data circuit bit error rate and data signaling rate.

1.7.5 Media access and highway transaction times. The manufacturer shall provide information relating average and maximum media access time and highway transaction time to highway configuration, loading, priority, and other relevant factors.

1.7.5.1 Media access time. An industrial control data highway shall have a maximum media access time of no more than **10 millisec** for a frame submitted at the highest priority by an arbitrary station under any set of conditions equivalent to those listed below.

- 2000 m highway length
- 1 X 10^6 bits/sec data signaling rate
- 20 stations participate in the token ring
- 10 stations initiate SDA messages with an average user data length of 16 octets simultaneously
- address length = 16 bits
- no more than one error occurs during each token rotation

One set of conditions under which this media access time may be achieved is given in Appendix 5-A.

At a given data rate, error rate, address length and propagation delay, the maximum media access time is determined by the number of token holding stations, the offered load at the highest priority, and the maximum frame length allowed.

1.7.5.2 Highway transaction time. The highway transaction time, when using the SDA or RDR services, is determined by the queuing delay within the initiating station, the media access time, the user data length and the number of retries required.

1.7.5.3 Definitions

- **Media access time** is defined as the period of time between a request becoming next to be transmitted and the time when the first bit of the SD for the corresponding request frame appears on the common bus medium.
- **Highway transmission time** is the period of time between submission of a request at the PLC-user interface and the appearance of the corresponding confirmation at that interface.

1.8 System availability

1.8.1 Effect of failures. No single failure of any part of any device used within or connected to the data highway shall cause failure of the entire control system, or of any function except those in which the failed device is directly involved.

It shall be possible to configure an industrial control system which can tolerate without the loss of communication function, changes of configuration, failure of any one transmission line, or failure of any one station.

1.8.2 Internal status and error reporting. The data highway shall have an internal status and error reporting capability.

1.8.3 Automatic recovery. The data highway shall be capable of automatic recovery after commonly occurring failures are corrected.

1.8.4 Control of stations. The data highway shall support loading, starting, stopping, reloading, and resetting of any station.

2 The interface to PROWAY

2.1 Organization

Part 2A specifies the data transfer services provided to the PROWAY user by the PROWAY Link Control sublayer of the Data Link (Highway) layer of the ISO reference model.

Part 2B specifies the administrative services provided to the PROWAY user by the station management entity.

2A. User-PLC interface and service specification

2A.1 Scope and field of application. This part specifies the data transfer services provided to the user of PROWAY (PLC user) by the PROWAY Link Control (PLC) sublayer at the boundary between the user of PROWAY and the PROWAY Link Control sublayer of the Data Link (Highway) layer of the ISO reference model. This standard specifies these services in an abstract way. It does not specify or constrain the implementation entities and interfaces within a computer system. The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 2A-1.

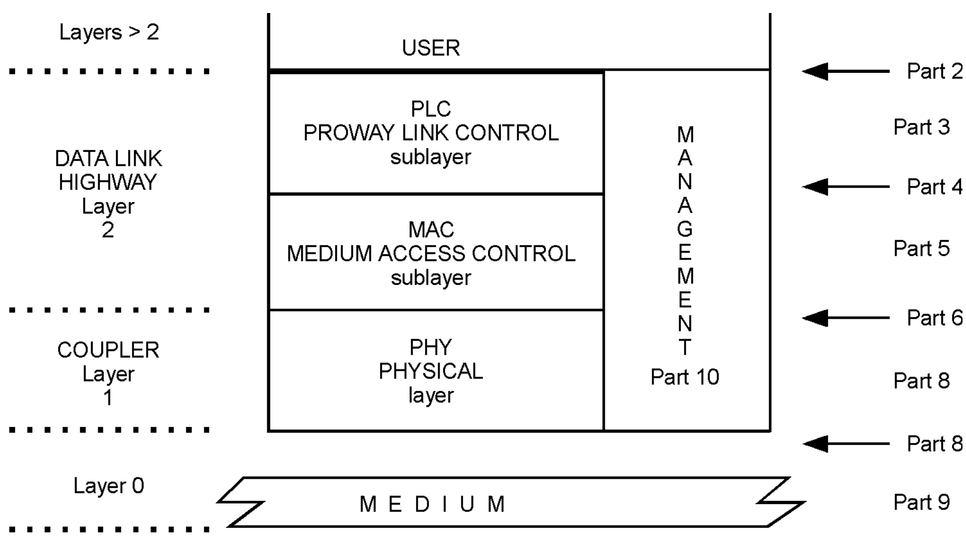


Figure 2A-1: Relationship to LAN model

2A.2 Overview of PROWAY services

2A.2.1 General description of services provided. This section describes informally the services provided to the user of PROWAY by the PROWAY Link Control (PLC) sublayer of the Highway layer. These services provide data transfer and control services between peer PLC entities. They provide the means by which PROWAY Link Control entities can exchange Link service data units (L_sdu) over the shared Process Data Highway, i.e., PROWAY Local Area Network (LAN). The data transfer can be open-loop or closed-loop, point-to-point or multipoint.

2A.2.2 Model used for the service specification. The general model and descriptive method are given in Reference 1, Appendix I-A. The specific application of this model to each service is given in 2A.4.

2A.2.3 Overview of services. The data transfer services provided to a user of PROWAY are:

- **Send Data with Acknowledge (SDA)**, compatible with the Single Frame service of Reference 3, Appendix I-A (when data is not requested).
- **Send Data with No Acknowledge (SDN)**, equivalent to and inter-operable with the Unacknowledged Connectionless service of Reference 3, Appendix I-A.
- **Request Data with Reply (RDR)**, compatible with the Single Frame service of Reference 3, Appendix I-A (when no data transmitted).

Note that a compatible Connection-Oriented service may be implemented in some stations on a PROWAY-LAN. This standard does not specify the Connection-Oriented service, which is specified in Reference 3, Appendix I-A.

The SDA and RDR services provide **confirmed** data transfers.

2A.2.3.1 Send Data with Acknowledge (SDA). This service allows a local user of PROWAY to send user-supplied data (Link service data unit, L_sdu) to a single remote station. The local user receives a confirmation of the receipt or nonreceipt of the data. At the remote station, this L_sdu (if received correctly) is passed to the remote user of PROWAY. If such an L_sdu is presented to a remote user, it is known to be correct and identical to the L_sdu presented at the local station. If an error in transmission occurs, the local PROWAY station will attempt to retransmit the data.

2A.2.3.2 Send Data with No Acknowledge (SDN). This service allows a local user of PROWAY to send user-supplied data (Link service data unit, L_sdu) to a single, a group of, or all remote stations. The local user receives confirmation of transmission completion but not of receipt of the data. At the remote station(s) this L_sdu (if received correctly) is passed to the remote user(s) of PROWAY. If such an L_sdu is presented to a remote user, it is known to be correct and identical to the L_sdu presented at the local station. However, there is no acknowledgment that such a delivery was made.

2A.2.3.3 Request Data with Reply (RDR). This service allows a local user of PROWAY to request data (Link service data unit, L_sdu) previously submitted by the user at a single remote station. The local user receives either the data requested or an indication that the data was not available. If the data is presented to the local user, it is known to be correct and identical to the L_sdu presented earlier at the remote station. In the event of transmission errors, the local PROWAY station will make additional requests for the data.

2A.2.4 Overview of interactions. These services are accomplished using a set of primitives. A request primitive is used by the local user to request a service. A confirmation primitive is returned to the local user when the service is completed. An indication primitive is passed to the remote user whenever an unsolicited event is noted. The primitives used for each of the above services are:

- **Send Data with Acknowledge (SDA)**
  L_DATA_ACK.request
  .indication
  .confirm
- **Send Data with No Acknowledge (SDN)**
  L_DATA.request
  .indication
  .confirm
- **Request Data with Reply (RDR)**
  L_REPLY.request
  .indication
  .confirm
  L_REPLY_UPDATE.request
  .confirm

The local user of PROWAY can have only one outstanding request (i.e., awaiting a confirmation) for each invocation of the local PLC entity supported by this station. A station must support one invocation of the local PLC entity for each priority and SSAP that it supports (see Section 3B.1.1).

2A.2.5 Overview of data highway operation. An overview of the combined operation of the Data Link layer (PLC and MAC sublayers) is given in Appendix 2-A.

2A.3 Mandatory features. The primitives which are mandatory for **INITIATOR** and **RESPONDER** stations are listed as M in the table below. Optional primitives are listed as O, and non-applicable primitives are listed as —.

	INITIATOR	RESPONDER
Send Data with Acknowledge (SDA)		
L_DATA_ACK.request	M	—
.indication	M	M
.confirm	M	—
Send Data with No Acknowledge (SDN)		
L_DATA.request	M	—
.indication	M	M
.confirm	O	—
Request Data with Reply (RDR)		
L_REPLY.request	M	—
.indication	M	M
.confirm	M	—
L_REPLY_UPDATE.request	O	M
.confirm	O	M

2A.4 Detailed specification of PLC interactions with the user of PROWAY

2A.4.1 Send Data with Acknowledge (SDA)

2A.4.1.1 Description of operation. The local PLC user prepares a Link service data unit (L_sdu) containing process information or commands, both transparent to PROWAY, for one remote PLC user. This data is passed to the local PLC entity by an L_DATA _ACK.request primitive over the local PLC-user interface. The local PLC entity accepts this service request and attempts to send the L_sdu to the remote PLC entity. The local PLC entity provides a confirmation to the local PLC user indicating the success or a failure of the data transfer.

A positive acknowledgment is required from the remote PLC entity before the local PLC entity returns a positive confirmation to the local PLC user. If this acknowledgment is not received in a timely manner, the local MAC entity will make up to a predefined number of attempts to pass the L_sdu to the remote MAC. No other traffic occurs on the local area network between the original transmission of the data and its associated acknowledgment.

When the frame is correctly received, the remote PLC entity receives and passes the L_sdu to the remote PLC user over its PLC-user interface using a L_DATA_ACK.indication primitive.

2A.4.1.2 Definition of primitives of local user-highway interface

2A.4.1.2.1 L_DATA_ACK._request

2A.4.1.2.1.1 Function. This primitive is the service request primitive for the send_data_with_ acknowledge data transfer service.

2A.4.1.2.1.2 Semantics. The primitive shall provide parameters as follows:

L_DATA_ACK.request
  (SSAP,
  DSAP,
  L_sdu,
  remote_address,
  service_class)

The SSAP and DSAP parameters specify the local and remote user link service access points involved in the data transfer service, as defined in Part 3C.

The remote_address parameter specifies the MAC address of the remote station, as defined in Part 5D. The remote_address must be an individual address.

The L_sdu parameter specifies the link service data unit to be transferred by the PLC entity.

The service_class parameter specifies the MAC access_class and thus the MAC priority desired for the data transfer, as defined in Part 5D. Recommended access_class assignments for PROWAY systems are as follows:

MAC Priority	Access_class	Usage
Highest	6	Urgent messages, i.e., those performing critical alarm, interlock, and control coordination functions
	4	Normal control actions and ring maintenance functions
	2	Routine data gathering and display, and data base update functions
Lowest	0	File and program transfers

2A.4.1.2.1.3 When generated. This primitive is passed from the local PLC user to the local PLC entity to request a L_sdu be sent to one remote PLC user using send_data_with_acknowledge procedures.

2A.4.1.2.1.4 Effect of receipt. Receipt of this primitive causes the local PLC entity to send an L_sdu using send_data_with_acknowledge procedures.

2A.4.1.2.1.5 Additional comments. Only one SDA, SDN, or RDR request may be concurrently outstanding (i.e., awaiting a confirmation) for each invocation, as defined in Part 3B, of the local PLC entity supported by this station.

2A.4.1.2.2 L_DATA_ACK.indication

2A.4.1.2.2.1 Function. This primitive is the service indication for the send_data_with_acknowledge data transfer service.

2A.4.1.2.2.2 Semantics. The primitive shall provide parameters as follows:

L_DATA_ACK.indication
  (SSAP,
  DSAP,
  local_address,
  remote_address,
  L_sdu,
  service_class)

The SSAP and DSAP parameters specify the local and remote user link service_access points involved in the data transfer service, as defined in Part 3C.

The local_address and remote_address parameters specify the source address and the destination address of the corresponding SDA frame, as defined in Part 5D.

The L_sdu parameter specifies the link service data unit of the corresponding SDA frame.

The service_class parameter specifies the actual MAC priority of the corresponding SDA frame, as defined in Part 5D.

2A.4.1.2.2.3 When generated. The primitive is passed from the remote PLC entity to the remote PLC user when a MA_DATA.indication with L_pdu_type = **SDA** is received and the received L_pdu is not a duplicate.

2A.4.1.2.2.4 Effect of receipt. The effect of receipt of this primitive on the remote PLC user is unspecified.

2A.4.1.2.2.5 Additional comments. The contents of the L_sdu parameter are logically complete and unchanged relative to the L_sdu parameter in the associated L_DATA_ACK.request primitive.

2A.4.1.2.3 L_DATA_ACK.confirm

2A.4.1.2.3.1 Function. This primitive is the service confirmation primitive for the send_data_with_acknowledge data transfer service.

2A.4.1.2.3.2 Semantics. The primitive shall provide parameters as follows:

L_DATA_ACK.confirm
  (SSAP,
  DSAP,
  remote_address,
  L_sdu,
  service_class,
  L_status)

The SSAP and DSAP parameters specify the local and remote user link service_access points involved in the data transfer service. They are identical to the SSAP and DSAP parameters of the corresponding L_DATA_ACK.request primitive.

The link service data unit is **null**.

The remote_address parameter specifies the destination_address parameter of the MA_DATA.confirmation primitive.

The service class parameter specifies the priority parameter of the MA_DATA.confirmation primitive.

The L_status parameter indicates the success or failure of the previous associated send_data_with_acknowledge data transfer request and whether any error condition is **temporary** or **permanent**.

The values that the Command_status components of the L_status parameter can assume are as follows:

Code	Meaning	Temporary/ Permanent
OK =	Command L_sdu accepted	__
TE =	No acknowledgment from remote station after specified retries (possibly due to a non-PROWAY station)	T
DS =	Local station disconnected from line	P
WD =	Local station watchdog timed out	P
IV =	Invalid parameters in request	P
RS =	Request for an unimplemented or unactivated service at remote DSAP: no action taken	P
LS =	Unimplemented service at local SSAP	P
UN =	Resources not available to remote PLC or MAC entity: no action taken	T
UE =	User-PLC interface error	P
PE =	Protocol error	T/P
IP =	Permanent implementation dependent error	P
IT =	Temporary implementation dependent error	T

In Reference 3, Appendix I-A, an additional Response_status component of L_status may be present.

2A.4.1.2.3.3 When generated. This primitive is passed from the local PLC entity to the local PLC user to indicate the success or failure of the previous associated send_data_with_acknowledge data transfer request.

2A.4.1.2.3.4 Effect of receipt. The effect of receipt of this primitive on the local PLC user is unspecified.

2A.4.1.2.3.5 Additional comments. If the transfer was unsuccessful, this primitive indicates that the remote PLC sublayer entity received and positively acknowledged the L_sdu. If an error in transmission occurs, the local MAC entity will make up to a predefined number of attempts to retransmit the data.

If L_status indicates a **Temporary** error, the local PLC-user entity may assume that a future retry of the associated request may be successful.

If L_status indicates a **Permanent** error, the local PLC-user entity should assume that management intervention may be required before a retry of the associated request may be successful.

It is assumed that sufficient information is available to the local PLC user to associate this confirmation with the corresponding request.

2A.4.2 Send Data with No Acknowledge (SDN)

2A.4.2.1 Description of operation. The local PLC user prepares data (process information or commands, both transparent to PROWAY) for any or all remote users. This data is passed by the L_DATA.request primitive over the PLC-user interface. The local PLC entity accepts this service request, attempts to send this data to the specified remote PLC entities, and returns a local transmission confirmation to the local PLC user. This confirmation reports only local failures.

There is no guarantee of delivery to the remote PLC entities addressed, since no remotely generated acknowledgments or local retries are employed. The data is transmitted once on the line and is received (subject to the propagation delay of the line) simultaneously by all addressed stations. Each remote PLC entity which receives the data passes it to the remote PLC user by an L_DATA.indication primitive.

2A.4.2.2 Inter-operability. Inter-operability, with any station conforming to References 3 and 4, Appendix I-A, is achieved using SDN procedures.

2A.4.2.3 Definition of primitives at local user-highway interface

2A.4.2.3.1 L_DATA.request

2A.4.2.3.1.1 Function. This primitive is the service request primitive for the send_data_with_no_acknowledge data transfer service.

2A.4.2.3.1.2 Semantics. The primitive shall provide parameters as follows:

L_DATA.request
  (SSAP,
  DSAP,
  remote_address,
  L_sdu
  service_class)

The SSAP and DSAP parameters specify the local and remote users' link service_access points involved in the data transfer service.

The remote_address parameter specifies the MAC address of the remote station as specified in Part 5D. The remote_address may be either an individual address, a group address, or a broadcast address.

The L_sdu parameter specifies the link service data unit to be transferred by the local PLC entity.

The service_class parameter specifies the MAC priority desired for the data transfer as specified in Part 5D.

MAC Priority	Access_class	Usage
Highest	6	Urgent messages, i.e., those performing critical alarm, interlock, and control coordination functions
	4	Normal control actions and ring maintenance functions
	2	Routine data gathering and display, and data base update functions
Lowest	0	File and program transfers

2A.4.2.3.1.3 When generated. This primitive is passed from the local PLC user to the local PLC entity to request that a L_sdu be sent to one, a group, or all remote users using send_data_with_no_acknowledge procedures.

2A.4.2.3.1.4 Effect of receipt. Receipt of this primitive causes the local PLC entity to send a L_sdu using send_data_with_no_acknowledge procedures.

2A.4.2.3.1.5 Additional comments. Only one SDA, SDN, or RDR request may be concurrently outstanding (i.e., awaiting a confirmation) for each invocation, as defined in Part 3B, of the local PLC entity supported by this station).

2A.4.2.3.2 L_DATA.indication

2A.4.2.3.2.1 Function. This primitive is the service indication primitive for the send_data_with_no_acknowledge data transfer service.

2A.4.2.3.2.2 Semantics. The primitive shall provide parameters as follows:

L_DATA.indication
(SSAP,
DSAP,
local_address,
remote_address,
L_sdu,
service_class)

The SSAP and DSAP parameters specify the local and remote user link service_access points involved in the data transfer.

The local address and remote_address parameters specify the source_address and destination_address of the corresponding SDN frame, as defined in Part 5D.

The L_sdu parameter specifies the link service data unit of the corresponding SDN frame.

The service_class parameter specifies the actual MAC priority parameter of the corresponding SDN frame, as defined in Part 5D.

2A.4.2.3.2.3 When generated. The primitive is passed from the remote PLC entity to the remote PLC user when a MA_DATA.indication L_pdu_type = **SDN** is received.

2A.4.2.3.2.4 Effect of receipt. The effect of receipt of this primitive on the remote PLC user is unspecified.

2A.4.2.3.2.5 Additional comments. The contents of the L_sdu parameter are logically complete and unchanged relative to the L_sdu parameter in the associated L_DATA.request primitive.

2A.4.2.3.3 L_DATA.confirm

2A.4.2.3.3.1 Function. This primitive is the confirmation of the send_data_with_no_acknowledge data transfer service.

2A.4.2.3.3.2 Semantics. The confirmation shall provide parameters as follows:

L_DATA.confirm
  (SSAP,
  DSAP,
  remote_address,
  service_class
  L_status)

The SSAP and DSAP parameters specify the local and remote user link service_access points involved in the data transfer service. They are identical to the SSAP and DSAP parameters of the corresponding L_DATA.request primitive.

The remote_address specifies the destination_address parameter of the MA_DATA.confirm primitive.

The service_class parameter specifies the priority parameter of the MA_DATA.confirm primitive.

The L_status parameter indicates the local success or failure of the previous associated send_data_with_no_acknowledge data transfer request.

The values that the Command_status component of the L_status parameter can assume are:

OK = Transmission complete at local station

DS = Local station disconnected from line

WD = Local station watchdog timed out

LE = Invalid parameters (locally detected)

LS = Unimplemented service at local station

2A.4.2.3.3.3 When generated. This confirmation is passed from the local PLC entity to the local PLC user to indicate the local success or failure of the previous associated send_data_with_no_acknowledge data transfer request.

2A.4.2.3.3.4 Effect of receipt. The effect of receipt of this confirmation by the local PLC user is unspecified.

2A.4.2.3.3.5 Additional comments. It is assumed that sufficient information is available to the local PLC user to associate this confirmation with the corresponding request.

2A.4.3 Request Data with Reply (RDR)

2A.4.3.1 Description of operation. The local PLC user requests data from a remote PLC user by a L_REPLY.request primitive passed to the local PLC entity over the PLC-user interface. The local PLC entity accepts this service request and sends a request for the data to the remote PLC entity. The local PLC entity provides a L_REPLY.confirm to the local PLC user with the requested data or a failure indication.

The remote PLC entity receives the request for data and immediately responds by transmitting a copy of data that had been previously submitted by the remote PLC user with an L_REPLY_UPDATE.request for the corresponding DSAP. The remote PLC user is informed of this reply transmission by an L_REPLY indication.

If a response is not received in a timely manner, the local MAC entity may make up to a predefined number of attempts to obtain the requested data. Between the original transmission of the request and the associated response, no other traffic occurs on the local area network.

The remote PLC user is responsible for maintaining valid data available to the remote PLC entity. The remote PLC user may make a L_REPLY_ UPDATE.request to update the shared_data_area, and a L_REPLY_ UPDATE.confirm is returned by the remote PLC after the update is complete. Note that the data updating procedures occur asynchronously to any data transfer requests.

2A.4.3.2 Definition of primitives at local user-PLC interface

2A.4.3.2.1 L_REPLY.request

2A.4.3.2.1.1 Function. This primitive is the service request primitive for the request_data_with_reply (RDR) service.

2A.4.3.2.1.2 Semantics. The primitive shall provide parameters as follows:

L_REPLY.request
(SSAP,
DSAP,
remote_address,
L_sdu,
service_class)

The SSAP parameter specifies the link service_access point of the local user requesting the data.

The DSAP parameter specifies which shared_data_area is requested from the remote PLC and thus which remote user will receive the indication primitive.

The remote address parameter specifies the MAC address of the remote station, as defined in Part 5D.

The link service data unit is **null**.

The service_class specifies the MAC priority of the request as specified in Part 5D. Recommended access class assignments for PROWAY systems are as follows:

MAC Priority	Access_class	Usage
Highest	6	Urgent messages, i.e., those performing critical alarm, interlock, and control coordination functions
	4	Normal control actions and ring maintenance functions
	2	Routine data gathering and display, and data base update functions
Lowest	0	File and program transfers

2A.4.3.2.1.3 When generated. This primitive is passed from the local PLC user to the local PLC entity to request data from one remote PLC entity.

2A.4.3.2.1.4 Effect of receipt. Receipt of this primitive causes the local PLC entity to request the data specified by the DSAP parameter using the request_data_with_response procedures.

2A.4.3.2.1.5 Additional comments. The remote PLC user is responsible for maintaining valid data available to the remote PLC entity in the shared_data_area. This data has been previously transferred to the remote PLC entity by an L _REPLY_UPDATE.request primitive with an identical DSAP parameter. The underlying PLC entity must prevent the remote PLC user and the remote PLC entity itself from accessing (one writes while the other reads) the shared_data_area at the same time.

Only one SDA, SDN, or RDR request may be concurrently outstanding (i.e., awaiting a confirmation) for each invocation, as defined in Part 3B, of the local PLC entity supported by this station.

2A.4.3.2.2 L_REPLY.indication

2A.4.3.2.2.1 Function. This primitive is the service indication primitive for the request_data_with reply service.

2A.4.3.2.2.2 Semantics. The primitive shall provide parameters as follows:

L_REPLY.indication
  (SSAP,
  DSAP,
  local_address,
  L_sdu,
  remote_address,
  service_class)

The SSAP parameter specifies the link service_access point of the local PLC user requesting the data.

The DSAP parameter specifies the remote PLC-user link service_access point that receives this indication. It also specifies which shared_data_area was transmitted as a response to an RDR frame.

The local_address and remote_address parameters specify the source_address and destination_address of the corresponding RDR frame.

The link service data unit is **null**.

The service_class parameter specifies the actual MAC priority of the corresponding RDR frame, as defined in Part 5D.

2A.4.3.2.2.3 When generated. This primitive is passed by the remote PLC entity to the remote PLC user when a shared_data_area is transmitted as a response to an RDR frame.

2A.4.3.2.2.4 Effect of receipt. The effect of receipt of this primitive on the remote PLC user is unspecified.

2A.4.3.2.3 L_REPLY.confirm

2A.4.3.2.3.1 Function. This primitive is the service confirmation primitive for the request_data_with_reply service.

2A.4.3.2.3.2 Semantics. The primitive shall provide parameters as follows:

L_REPLY.confirm
  (SSAP,
  DSAP,
  remote_address,
  L_sdu,
  service_class,
  L_status)

The SSAP and DSAP parameters specify the local and remote user link service_access points involved in the data transfer service. They are identical to the SSAP and DSAP parameters of the corresponding L_REPLY.request primitive. In addition the DSAP identifies the remote shared_data_area whose contents are the L_sdu that accompanies this confirmation.

The remote_address specifies the destination_address parameter of the MA_DATA.confirmation primitive.

The L_sdu parameter identifies the data returned by the remote PLC entity.

The service_class parameter specifies the priority parameter of the MA_DATA.confirmation primitive.

The L_status parameter indicates the success or failure of the previous associated request_data_with_response data transfer request and whether any error condition is temporary or permanent.

The values that the Response_status component of the L_status parameter can assume are follows:

Code	Meaning	Temporary/ Permanent
OK =	Response L_sdu accepted	—
TE =	No response from remote station after specified retries (possibly due to a non-PROWAY station)	T
DS =	Local station disconnected from line	P
WD =	Local station watchdog timed out	P
IV =	Invalid parameters in request	P
RS =	Request for an unimplemented or unactivated service at remote DSAP: no action taken	P
LS =	Unimplemented service at local SSAP	P
UN =	Resources not available to remote PLC or MAC entity: no action taken	T
UE =	User-PLC interface error	P
PE =	Protocol error	T/P
IP =	Permanent implementation dependent error	P
IT =	Temporary implementation dependent error	T
NE =	Response L_sdu never submitted at destination	P

In Reference 3, Appendix I-A, an additional Command_status component of L_status may be present.

2A.4.3.2.3.3 When generated. This primitive is passed from the local PLC entity to the local PLC user to indicate the success or failure of the previous associated request_data _ with_reply data transfer request and to pass the requested data if the transfer was successful.

2A.4.3.2.3.4 Effect of receipt. The effect of receipt of this primitive on the local PLC user is unspecified.

2A.4.3.2.3.5 Additional comments. The local PLC entity delivers either the requested data or the reason for failure to the local PLC user. If an error in transmission occurs, the local MAC entity will make up to a predefined number of requests for the data.

If L_status indicates a **Temporary** error, the local PLC-user entity may assume that a future retry of the associated request may be successful.

If L_status indicates a **Permanent** error, the local PLC-user entity should assume that management intervention may be required before a retry of the associated request may be successful.

It is assumed that sufficient information is available to the local PLC user to associate this confirmation with the corresponding request.

2A.4.3.2.4 L_REPLY_UPDATE.request

2A.4.3.2.4.1 Function. This primitive is the update_request primitive of the request_data_with_ reply service.

2A.4.3.2.4.2 Semantics. The primitive shall provide parameters as follows:

L_REPLY_UPDATE.request
  (DSAP,
  L_sdu)

The DSAP parameter specifies the remote PLC-user link service_access point making this update_ request and which shared_data_area is to be updated.

The L_sdu parameter specifies the new contents of the shared_data_area identified by the DSAP parameter.

2A.4.3.2.4.3 When generated. The primitive is passed by the remote PLC user to the remote PLC entity to request update of a shared_data_area.

2A.4.3.2.4.4 Effect of receipt. Receipt of this primitive causes the remote PLC entity to attempt to update the specified shared_data_area.

2A.4.3.2.4.5 Additional comments. The shared_data_area can be updated only if the remote PLC entity is not attempting to generate a response by accessing the same shared_data_area.

This primitive has significance only to the station containing the shared_data_area, i.e., the remote station.

2A.4.3.2.5 L_REPLY_UPDATE.confirm

2A.4.3.2.5.1 Function. This primitive is the update_confirmation primitive of the request_data_with_reply service.

2A.4.3.2.5.2 Semantics. This primitive shall provide parameters as follows:

L_REPLY_UPDATE.confirm
  (DSAP,
  L_status)

The DSAP parameter specifies the remote PLC-user link service_access point that receives this update_confirmation and which shared_data_area was the subject of the attempted update.

The L_status parameter specifies success or failure of the corresponding update_request. The values that L_status can assume are:

Code	Meaning
OK =	shared data area updated
TE =	shared data area busy and not updated

2A.4.3.2.5.3 When generated. This primitive is passed from the remote PLC entity to the remote PLC user to indicate success or failure of the corresponding L_REPLY_UPDATE.request.

2A.4.3.2.5.4 Effect of receipt. The effect of receipt of this primitive on the remote PLC user is unspecified.

2B. User-management interface and service specification

2B.1 Scope and field of application. This section specifies the administrative services related to layers 1 and 2 of the ISO reference model that are provided by the Management entity of each station at the PROWAY management interface between the user of PROWAY and that management entity. This standard specifies these services abstractly. It does not specify or constrain the implementation of entities or interfaces within a computer system. The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 2B-1.

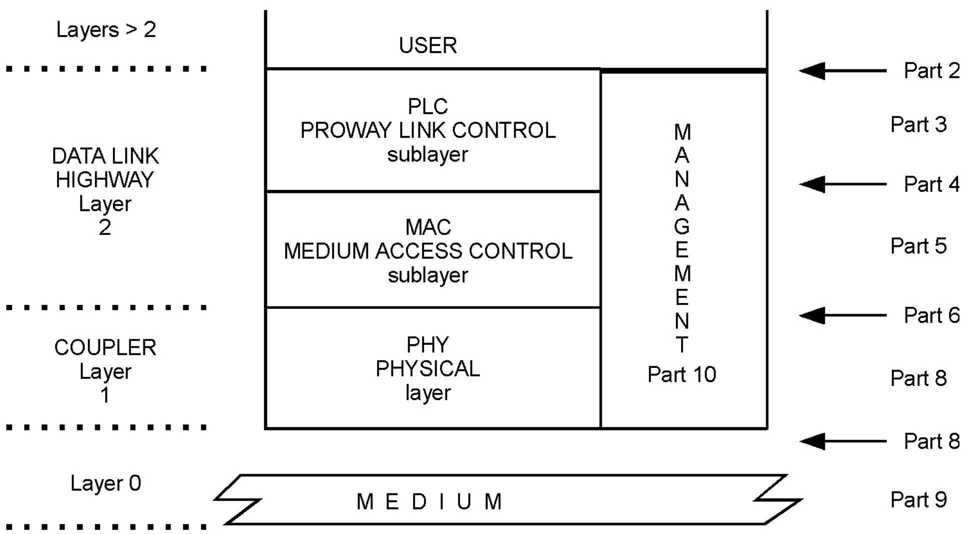


Figure 2B-1: Relationship to LAN model

2B.1.1 Mandatory features. The management functions described in Part 2B are mandatory if the corresponding PLC, MAC or PHY function is implemented. The specific primitives and codings given in Part 2B are subject to further study and harmonization with the work of the IEEE 802 committee.

2B.2 Overview of management services

2B.2.1 General description of services provided at the user-management interface

This section informally describes the L_MGMT services related to layers 1 and 2 of the ISO reference model provided to the user of PROWAY by each station's management entity. These services include the following:

1) changes in the configuration or status of the station

2) requests for information on token ring performance or membership

3) notifications of unsolicited changes in station or token ring status

4) request for the current value of PLC, MAC, or PHY entity parameters

The L_MGMT service has only local significance.

2B.2.2 Model used for the service specification. The model and descriptive method are given in Reference 1, Appendix I-A.

2B.2.3 Overview of interactions. One set of primitives is provided at the user_management interface:

L_MGMT.request
.confirm
.indication

L_MGMT.request primitive is passed from the local user of PROWAY to the local management entity to request a change in the station's configuration or status or current information on token ring performance or membership.

L_MGMT.indication primitive is passed from the local management entity to a designated local user to indicate a change in station status or token ring membership.

L_MGMT.confirm primitive is passed from the local management entity to the local user of PROWAY to convey the results of the corresponding L_MGMT.request primitive and any requested information.

2B.3 Detailed specification of management interactions with the user of PROWAY

2B.3.1 L_MGMT.request

2B.3.1.1 Function. This primitive is the service request primitive of the L_MGMT service.

2B.3.1.2 Semantics. The primitive shall provide parameters as follows:

L_MGMT.request
(SSAP,
code_type,
arguments)

SSAP specifies the link service_access point of the entity requesting the corresponding L_MGMT service.

The code_type parameter can assume any one of the values shown in Table 2-1.

The arguments are dependent upon the code_type chosen and are listed in Table 2-1 along with the corresponding code_type. The allowed format and range of values for each parameter are specified in Part 10B, 10C, or 10D, as appropriate.

2B.3.1.3 When generated. This primitive is passed from the local user of PROWAY to the local management entity to request a management action.

2B.3.1.4 Effect of receipt. Receipt of this primitive by the management entity causes the management entity to perform the requested action. This action may require the cooperation of the PLC, MAC, or PHY entity.

2B.3.1.5 Additional comments

2B.3.2 L_MGMT.confirm

2B.3.2.1 Function. This primitive is the service confirmation primitive of the L_MGMT service.

2B.3.2.2 Semantics. This primitive shall provide parameters as follows:

Table 2-1: Management request codes and arguments

Code_Type	Action Requested	Arguments
1	Return list of all stations participating in the token ring	None
2	Return value of PLC, MAC, or PHY parameter	L_user identifier, Parameter identifier, access_control information
3	Activate LSAP (except for RDR response component	SSAP, service activated at this SSAP, role in each service activated, maximum L_sdu length for each service activated
4	Activate RSAP (RDR response component of this LSAP	SSAP to receive L_REPLY.indications when a RDR request is received at this DSAP, DSAP associated with this shared_buffer_area, shared buffer_area identification (format is implementation dependent)
5	Deactivate SAP	SAP
6	Enter line_disconnect state	None
7	Enter line_ connect state	None
8	Enter configure state	None

L_MGMT.confirmation
(SSAP,
local_address,
code_type,
results,
Mgt_status)

SSAP specifies the link service_access point of the local user of PROWAY involved in the corresponding L_MGMT service.

Local_address specifies this station's MAC address as defined in Part 5D.

Code_type can assume any of the values indicated in Table 2-2.

Results provide information dependent on the code_type as listed in Table 2-2. The allowed range of values for each parameter is specified in Part 10B, 10C, or 10D, as appropriate.

Table 2-2: User-management confirmation codes and results

Code_Type	Action Attempted	Results
1	Return list of all stations participating in the token ring	Active_stations list. (format is implementation dependent)
2	Value of specified PLC, MAC, or PHY layer parameter	Status, parameter_value
3	Activate LSAP	Services activated at this SSAP, role in each active service, maximum L_sdu length for each service activated, shared buffer area identification, user to receive L_REPLY.indication
5	Deactivate SAP	SAP
6	Entry line_disconnect state	None
7	Entry line_ connect state	None
8	Entry configure state	None

L_MGT_status indicates success or failure of the action requested in the corresponding L_ MGMT.request. Mgt_status may assume the following values:

Code	Meaning
OK =	Requested action performed
CE =	Unimplemented code type
IP =	Invalid parameters
RJ =	Unable to perform requested action

2B.3.2.3 When generated. This primitive is passed from the local management entity to the requesting entity after the requested management operation is complete.

2B.3.2.4 Effect of receipt. The effect of receipt of this primitive on the user of PROWAY is unspecified.

2B.3.2.5 Additional comments. This primitive has local significance only.

2B.3.3 L_MGMT.indication

2B.3.3.1 Function. This primitive is the indication primitive of the L_MGMT service.

2B.3.3.2 Semantics. The primitive shall provide parameters as follows:

L_MGMT.indication
(SAP,
code_type)

SAP is the user notification service_access point of the entity that receives this service indication as defined in 3C.2.3.

Code_type specifies the change that has occurred. Table 2-3 itemizes each reported change.

Reason indicates the underlying cause of L_MGMT.indication.

Table 2-3: User-Management indication codes and reason

Code_Type	Associated Change	Reason
1	A change has occurred to the active stations list.	Defined in 10E
2	The station has entered the line_disconnect state spontaneously.	Defined in 10A

Additional changes to be reported are under study.

2B.3.3.3 When generated. This primitive is passed from the local management entity to the entity identified by the user_notification_SAP after the management entity has been notified of a change by the PLC, MAC or PHY layer.

2B.3.3.4 Effect of receipt. The effect of receipt of this primitive on the receiving entity is unspecified.

Appendix 2-A Overview of data highway operation

2-A.1 Organization of data highway overview. Figures 2-2 and 2-3 show the overall diagrams of the local and remote highway entities of a station that supports a single invocation of the local PLC entity. These figures show the combined actions of the PLC and MAC sublayers and the PHY layer as seen from the perspective of the local and remote users of PROWAY.

Figures 2-4 through 2-21 describe the operation of each service provided by the PROWAY data highway.

The description of each of the services begins with a topological and a sequential diagram which depicts the operation of the entire PROWAY data highway from the perspective of the PROWAY users. They are followed by example state diagrams for the local and remote users of this service. (These diagrams are included to clarify PROWAY_user interactions. They are not referenced in this standard.) The local and remote highway entity state diagrams show the operation of the Data Link (Highway) layer (PLC and MAC sublayers combined) as seen by the users of PROWAY. These diagrams show specific PLC-user and PLC-MAC interactions for this service.

The specifics of the PLC sublayer are given in Part 3, while the specifics of the MAC sublayer are given in Part 5.

In all highway state diagrams, the PLC states are identical to those of the same name defined in Part 3. In the topological diagrams, the solid arcs within PROWAY represent logical paths and the dotted lines represent frame transmissions. The value of the L_status parameter in a configuration depends on the logical path taken. In sequential diagrams, the time axis is shown as a pair of vertical lines with time increasing toward the bottom of the page and a malfunction indicated by an "X." All diagrams given in Appendix 2-A are for explanation only and are not a requirement of this standard. In case of conflict between Appendix 2-A and other parts of this standard, the other parts take precedence.

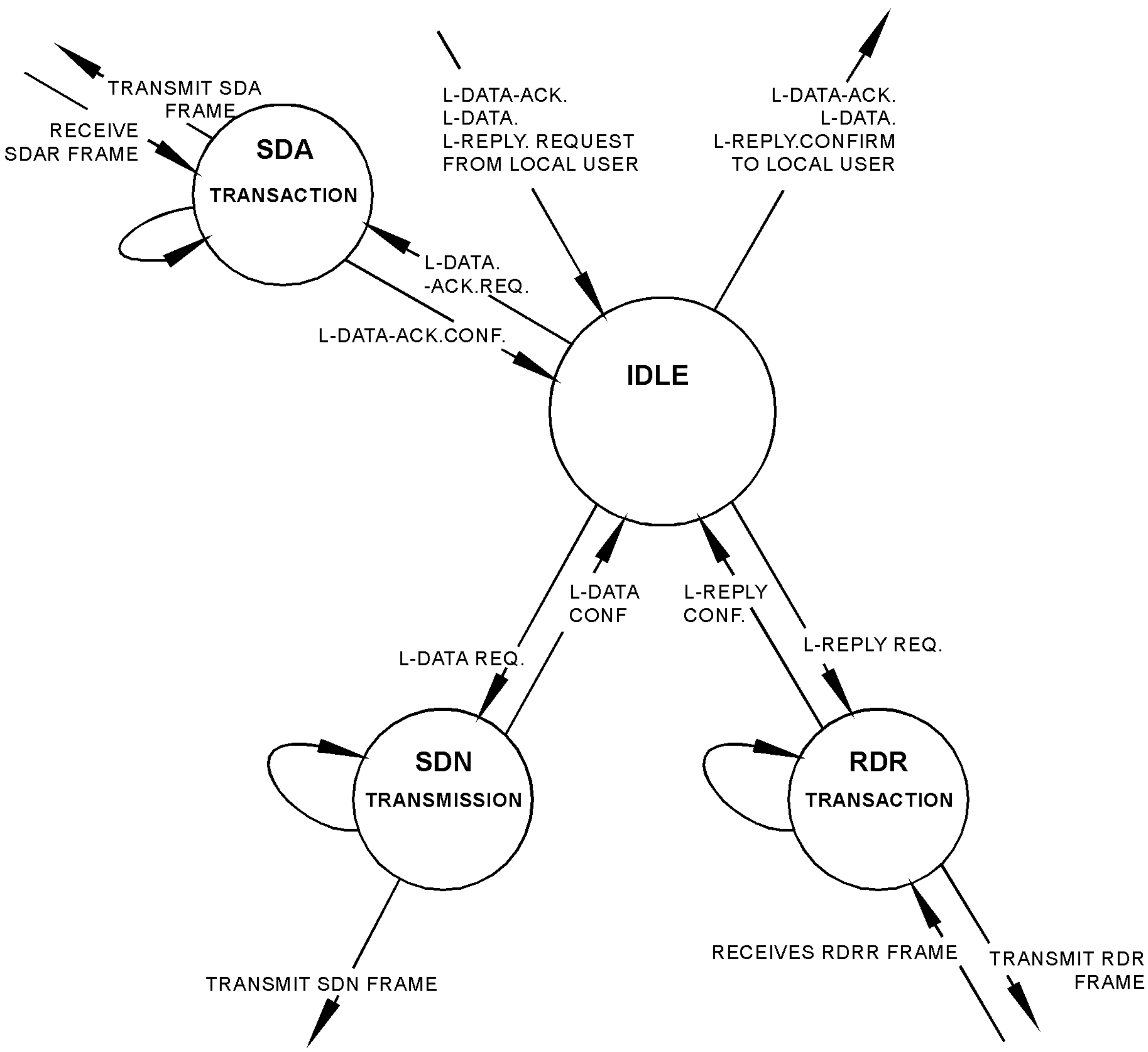


Figure 2-2: Local station interaction diagram

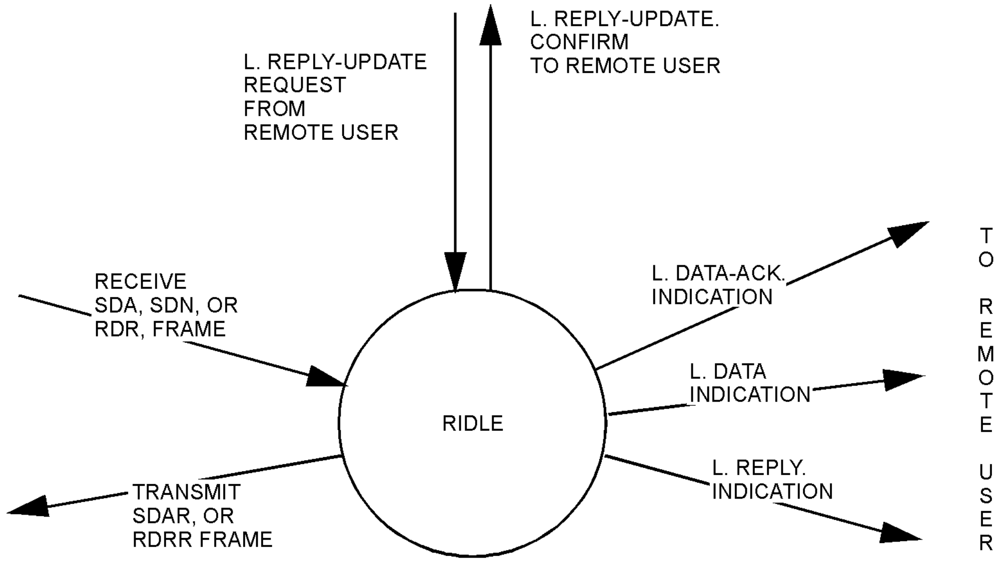


Figure 2-3: Remote station interaction diagram

2-A.2 Send Data with Acknowledge (SDA) service diagram

2-A.2.1 Topological and sequential relationships. The topological behavior of the send_data_with_acknowledge service is shown in Figure 2-4.

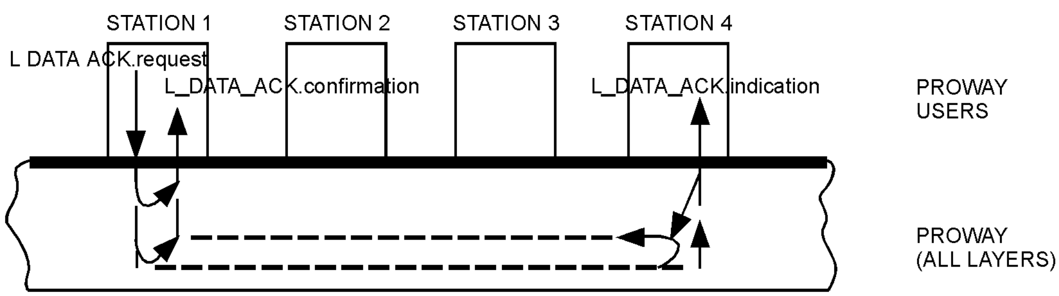


Figure 2-4: Topological behavior of the Send_Data_with_Acknowledge service

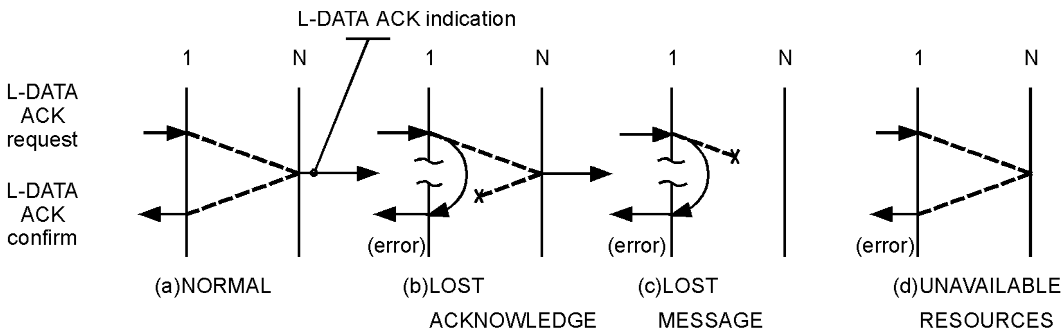


Figure 2-5: Sequential relationship of the Send_Data_with_Acknowledge service

2-A.2.2 Service state diagrams. The state diagrams for the send_data _with_acknowledge service are shown in Figures 2-6 through 2-9. Figures 2-6 and 2-7 show local and remote PLC-user state diagrams which indicate how PLC users interact at the PLC-user interface. Figures 2-8 and 2-9 show local and remote highway state diagrams to indicate how PROWAY processes the user service requests. Figures 2-8 and 2-9 correspond to sections of the overall highway entity state machines shown in Figures 2-2 and 2-3.

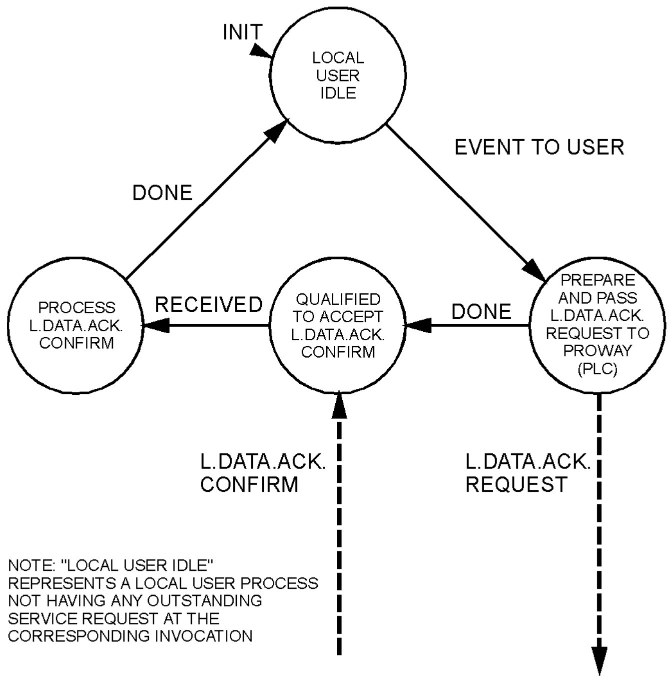


Figure 2-6: Local user state diagram for the Send_Data_with_Acknowledge service

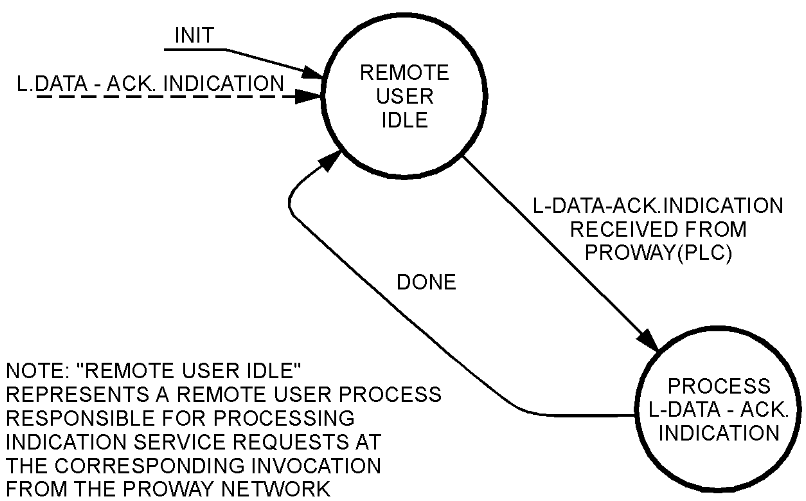


Figure 2-7: Remote user state diagram for the Send_Data_with_Acknowledge service

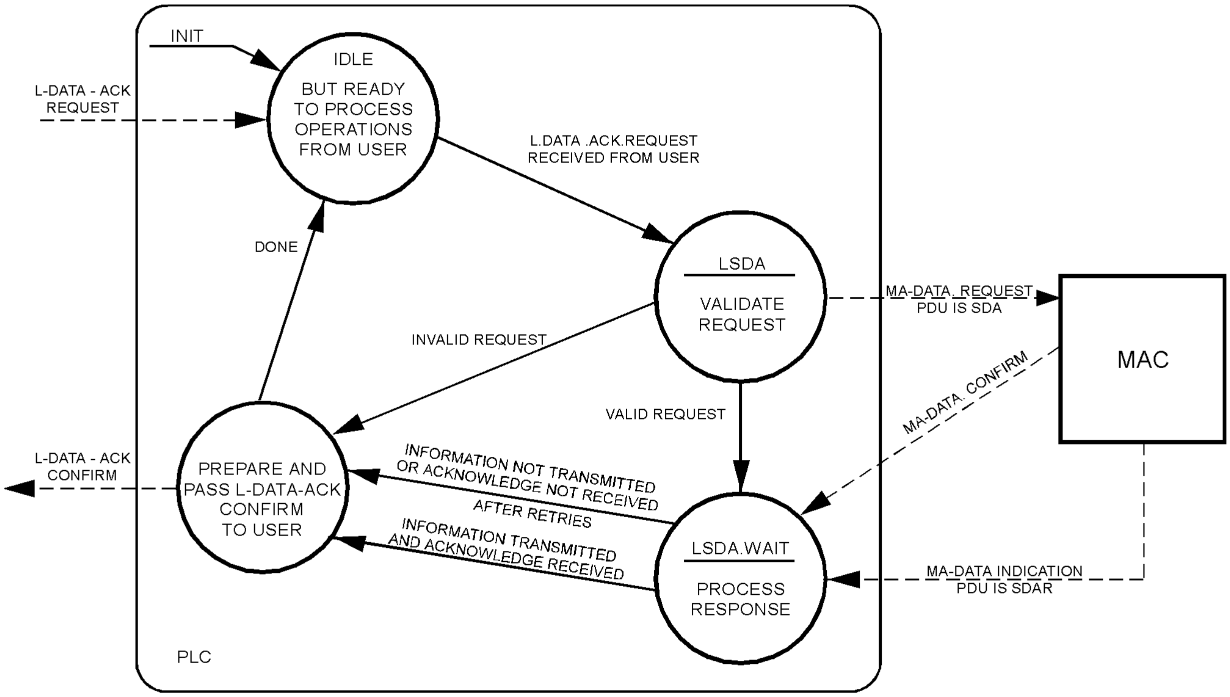


Figure 2-8: Local highway entity state diagram of the Send_Data_with_Acknowledge service

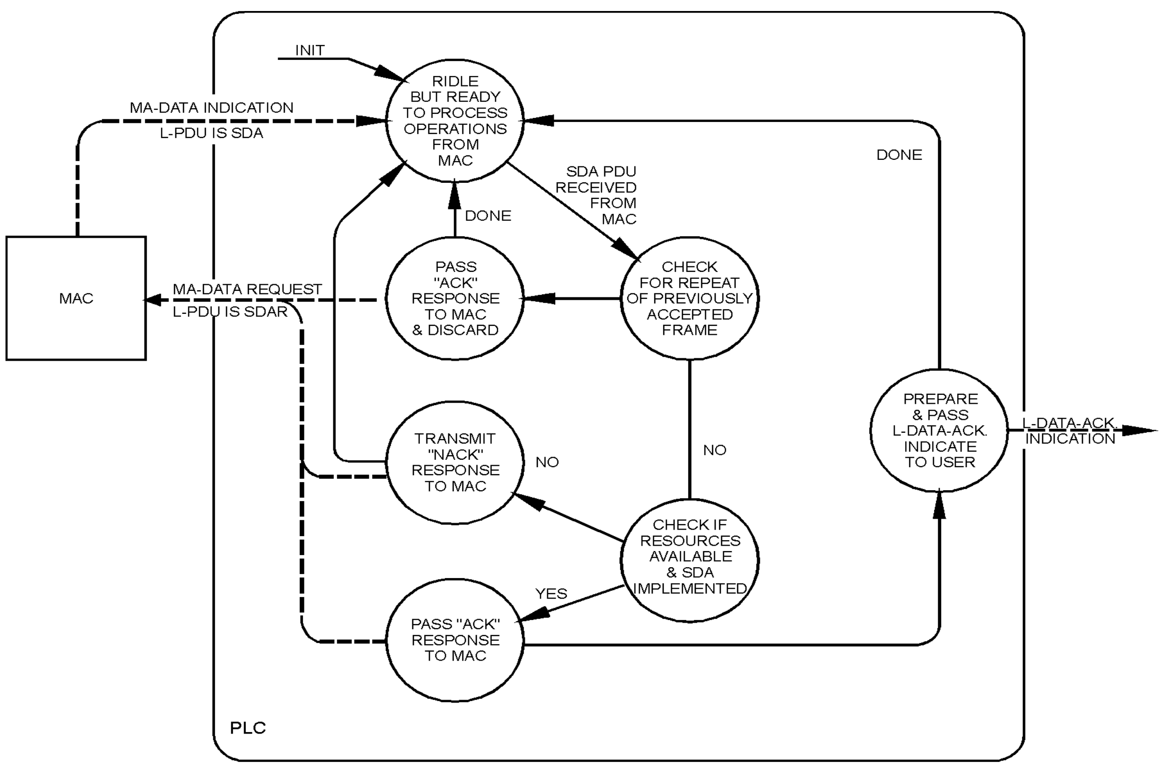


Figure 2-9: Remote highway entity state diagram for the Send_Data_with_Acknowledge service

2-A.3 Send Data with No acknowledge (SDN) service diagram

2-A.3.1 Topological and sequential relationships. The topological behavior of the send_data_with_no_acknowledge service is shown in Figure 2-10.

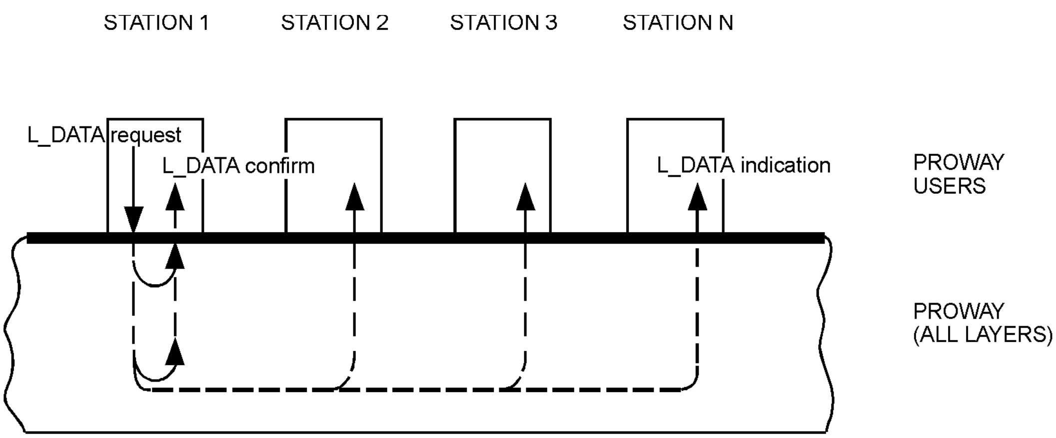


Figure 2-10: Topological behavior of the Send_ Data_with_No_acknowledge service

The sequential relationship of the send_data_with_no_acknowledge service is shown in Figure 2-11.

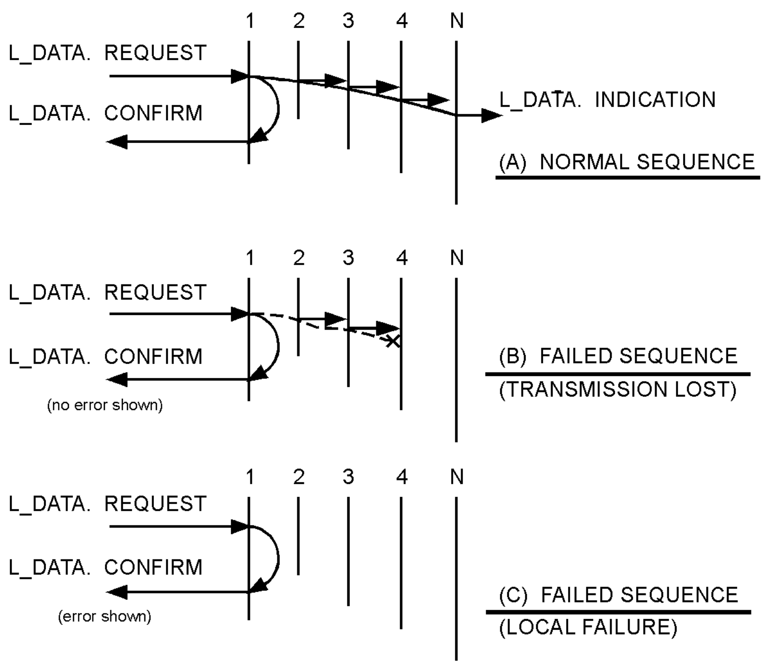


Figure 2-11: Sequential relationship of the Send_Data_with_No_acknowledge service

2-A.3.2 Service state diagrams. The state diagrams for the send _ data _with_no_acknowledge service are shown in Figures 2-12 through 2-15. Figures 2-12 and 2-13 show local and remote PLC-user state diagrams to indicate how the PLC-user interacts at the PLC-user interface. Figures 2-14 and 2-15 show local and remote highway entity state diagrams to indicate how PROWAY processes the user service requests. These last diagrams correspond to sections of the overall highway entity state machines shown in Figures 2-2 and 2-3.

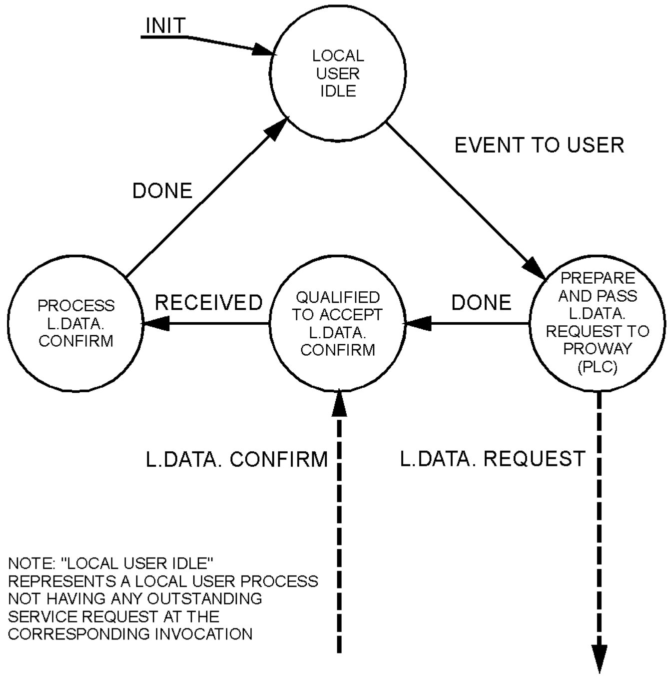


Figure 2-12: Local user state diagram for the Send_Data_with_No_acknowledge service

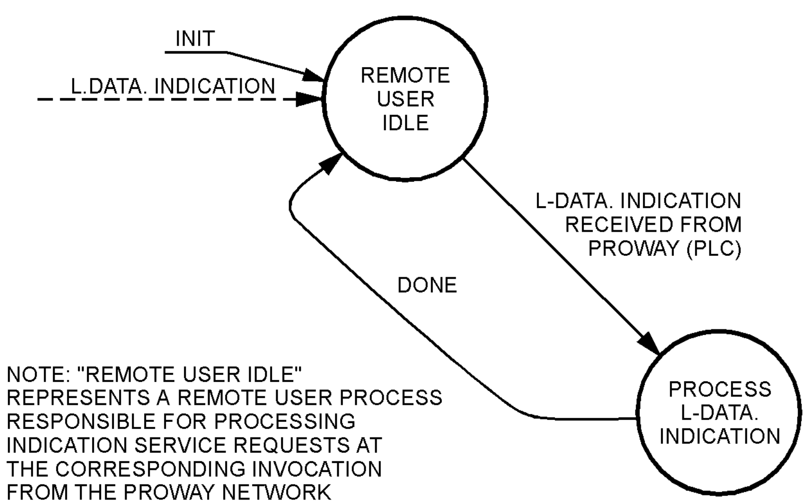


Figure 2-13: Remote user state diagram for the Send_Data_with_No_acknowledge service

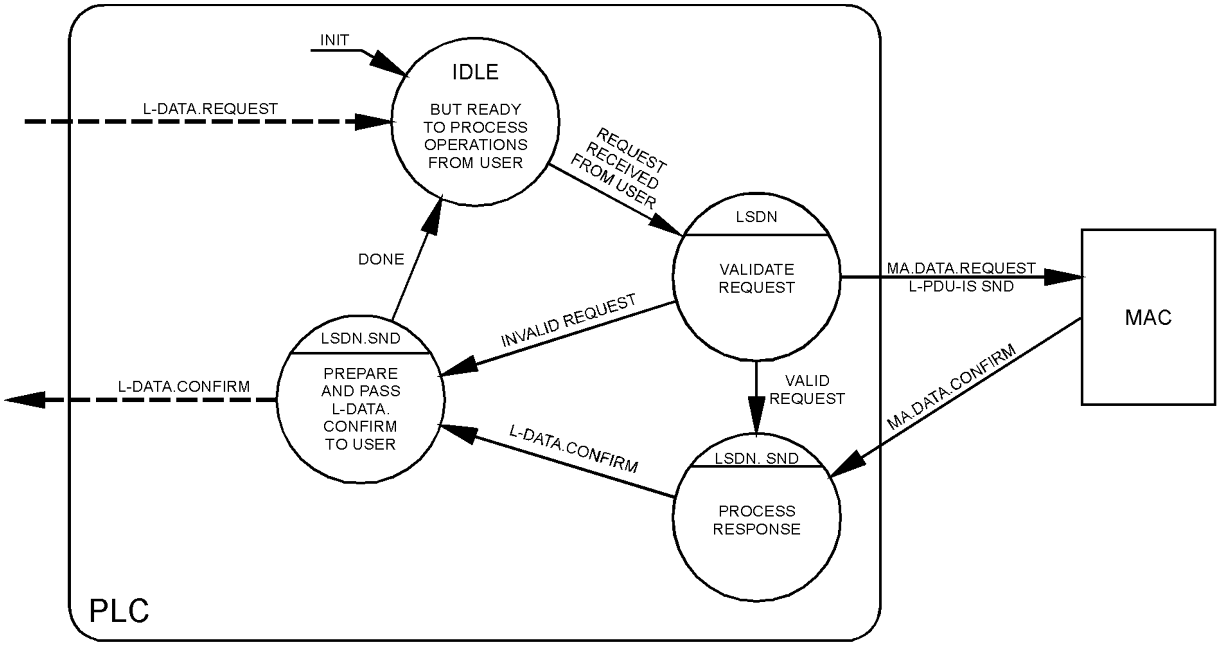


Figure 2-14: Local highway entity state diagram for the Send_Data_with_No_acknowledge service

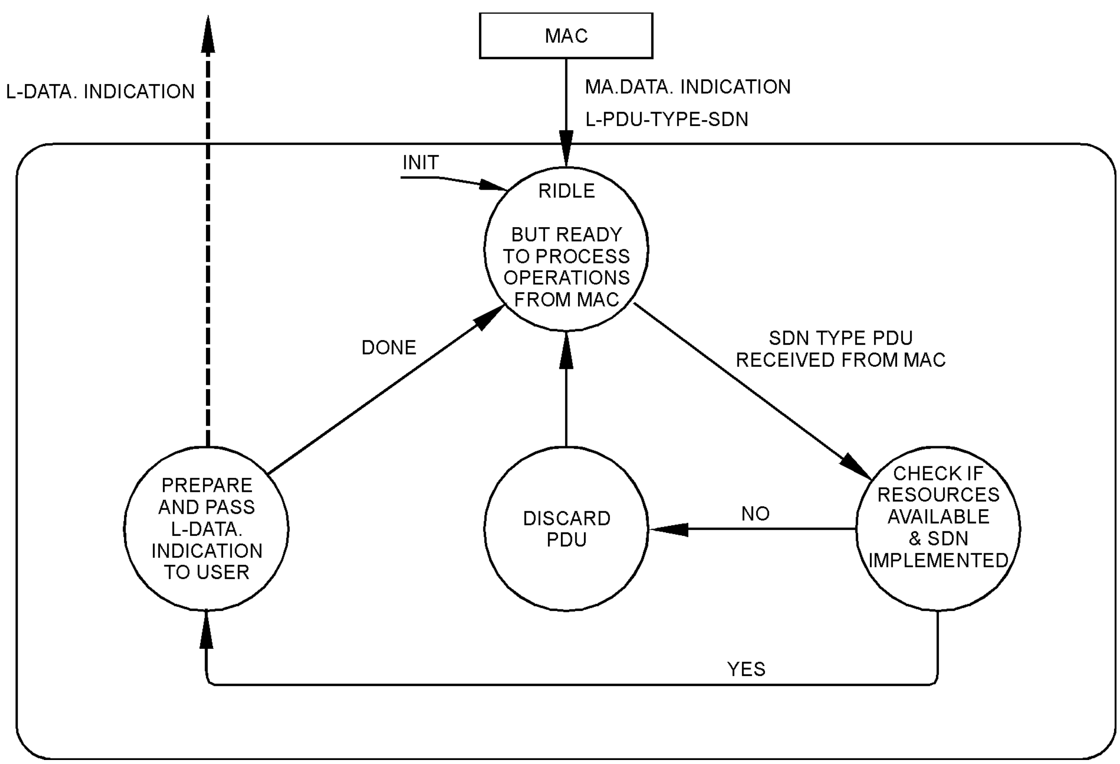


Figure 2-15: Remote highway entity state diagram for the Send_Data_with_No_acknowledge service

2-A.4 Request Data with Reply (RDR) service diagram

2-A.4.1 Topological and sequential relationships. The topological behavior of the Request_Data_With_Reply service is shown in Figure 2-16.

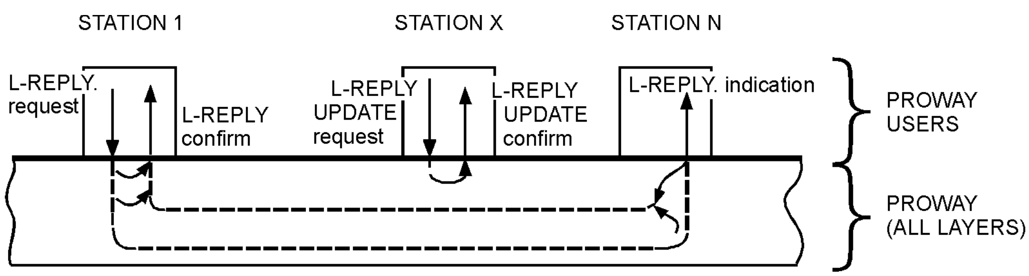


Figure 2-16: Topological behavior of the Request_Data_with_Reply service

The sequential relationship of the request_data_with_reply service is shown in Figure 2-17.

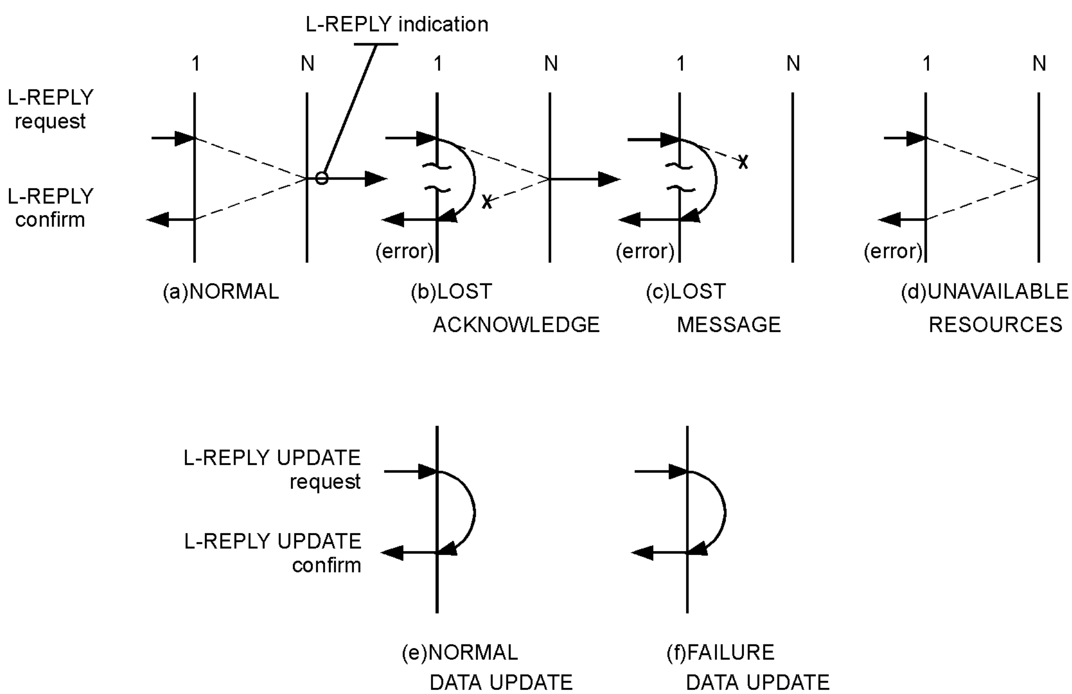


Figure 2-17: Sequential relationship of the Request_Data_with_Reply service

2-A.4.2 Service state diagrams. The state diagrams for the request_data_with_reply service are shown in Figures 2-18 through 2-21. Figures 2-18 and 2-19 show local and remote PLC-user state diagrams to indicate how the user interacts at the PLC-user interface. Figures 2-20 and 2-21 show local and remote highway entity state diagrams to indicate how PROWAY processes the user service requests. These last diagrams correspond to sections of the overall highway entity state machines shown in Figures 2-2 and 2-3.

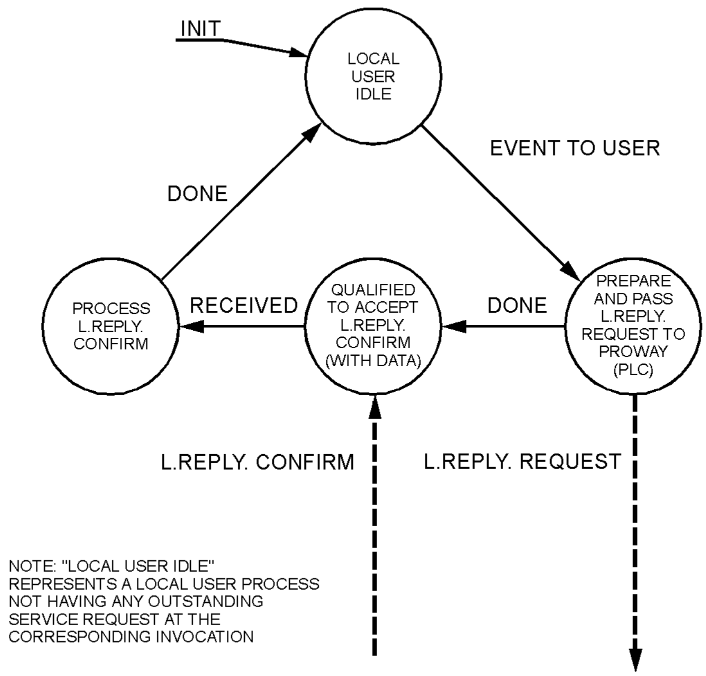


Figure 2-18: Local user state diagram for the Request_Data_with_Reply service

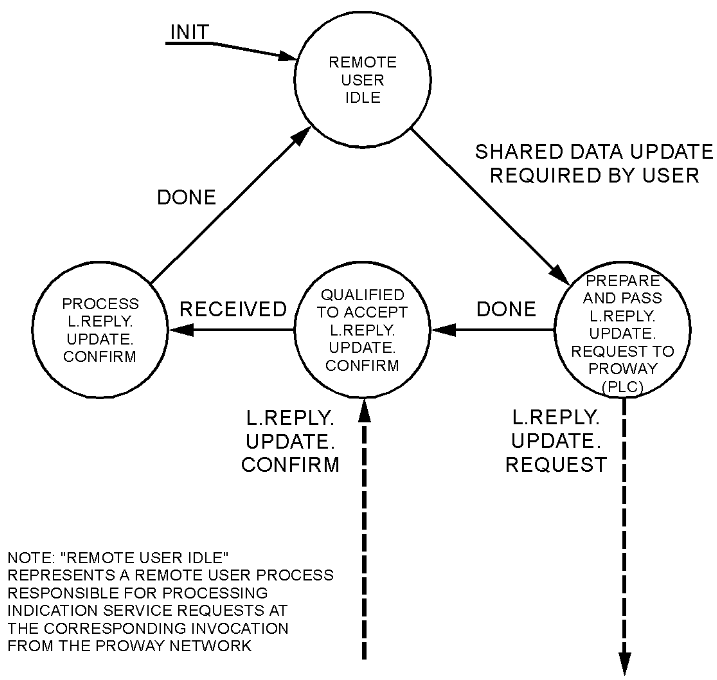


Figure 2-19: Remote user state diagram for the Request_Data_with_Reply service

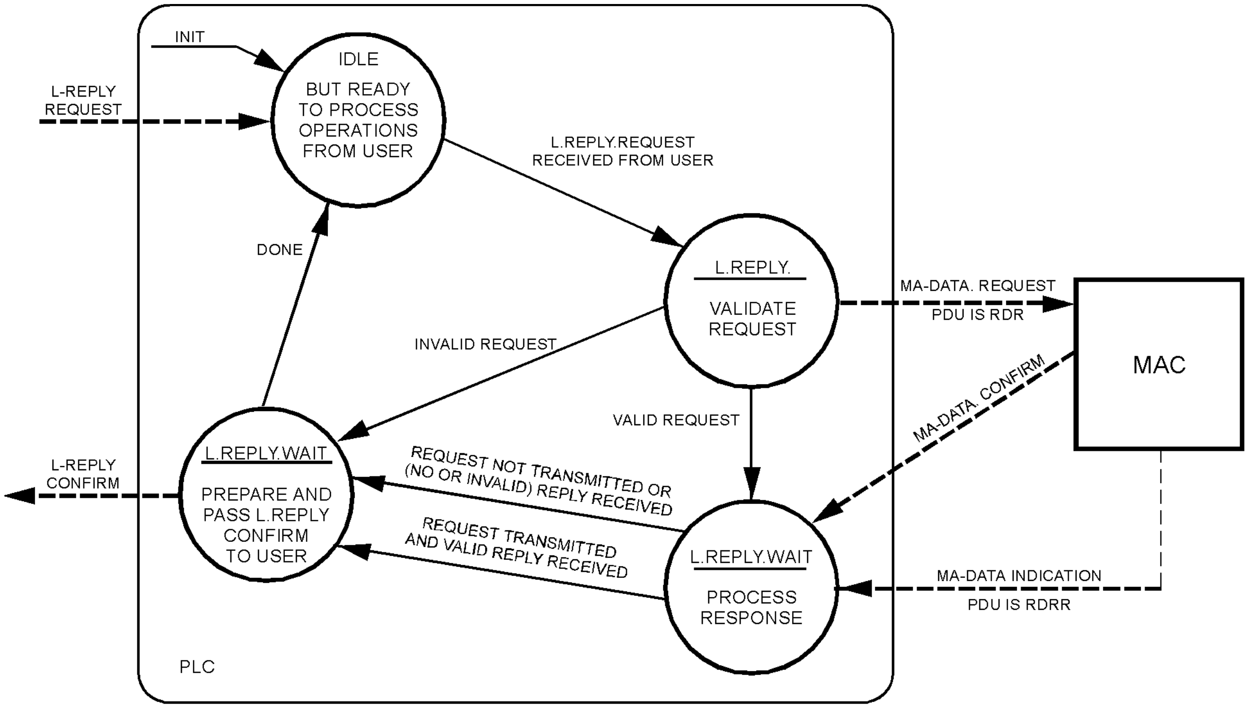


Figure 2-20: Local highway unit state diagram for the Request_Data_with_Reply service

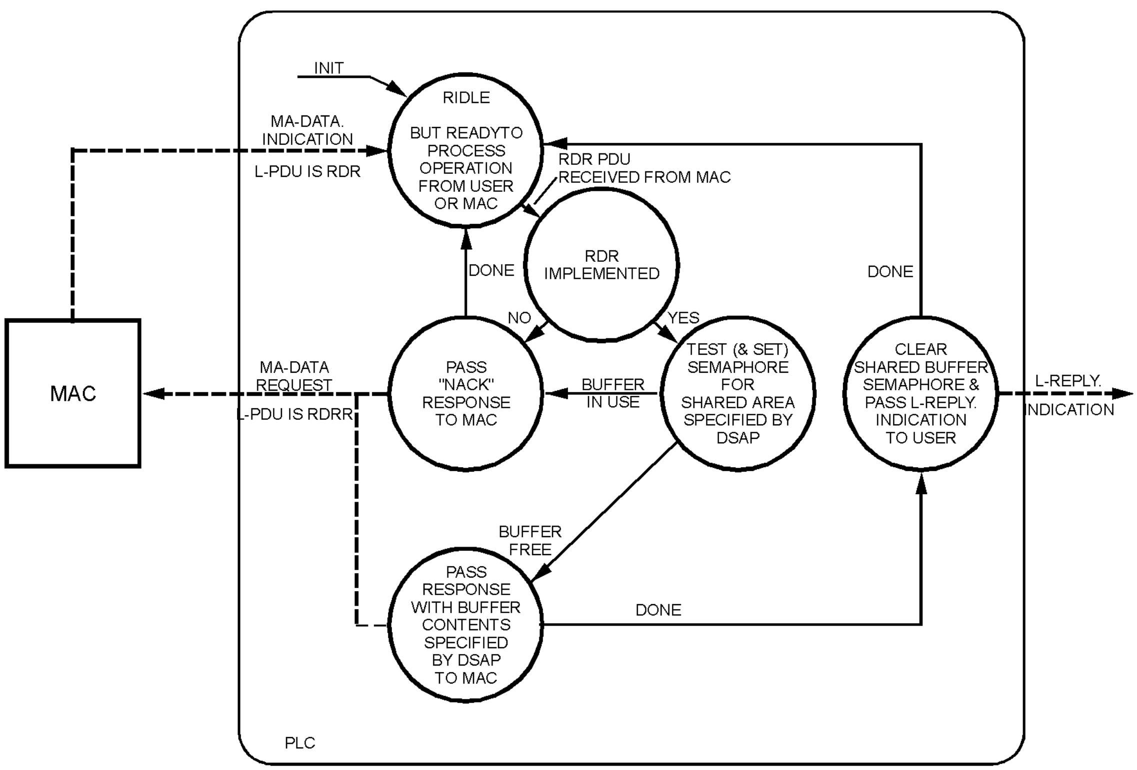


Figure 2-21: Remote highway entity state diagram for the Request_Data_with_Reply service data transmission

3 The PROWAY Link Control (PLC) sublayer

Part 3 specifies the operation of the PROWAY Link Control (PLC) sublayer of the Data Link (Highway) layer of the ISO reference model in an abstract way. It does not specify or constrain the implementation of PLC sublayer entities or interfaces within a computer system.

An introduction to the interactions of the PLC sublayer with the user of PROWAY and with the MAC sublayer of the Data Link layer is given in Appendix 2-A.

Part 3A gives precise definitions of terms and procedures used in the PLC machine state tables and specifies mandatory features of the PLC mechanism.

Part 3B, PLC state machines and descriptions, gives the formal definition of the PLC local and remote state machines using state transition tables and state diagrams.

Part 3C defines the PLC Link_protocol data unit (L_pdu) by which a PLC entity communicates with cooperating PLC entities.

The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 3-1.

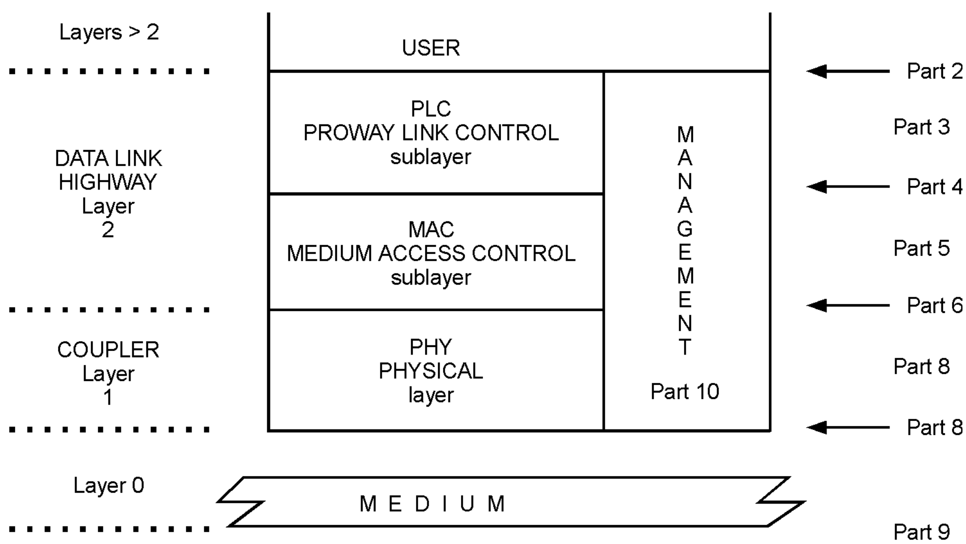


Figure 3-1: Relationship to LAN model

3A. PROWAY Link Control (PLC) sublayer definitions and mandatory features

3A.1 Notations used in PLC machine state tables

3A.1.1 Notations used for User-PLC interface parameters. This section defines abbreviations used in the PLC machine state tables for primitives and parameters passed across the user_PLC interface. For precise definitions, see Part 2A.

SSAP = local user link service_access point parameter in a primitive at the user-PLC interface

LA = local address parameter in a primitive at the user-PLC interface

DSAP = remote user link service_access point parameter in a primitive at the user-PLC interface

RA = remote address parameter in a primitive at the user-PLC interface

SC = service_class parameter in a primitive at the user-PLC interface

L_Status = L_status parameter in a primitive at the user-PLC interface

L_sdu = the link service data unit parameter in a primitive at the user-PLC interface

Confirm = confirmation

3A.1.2 Notations used for Link_protocol data unit parameters. This section defines abbreviations used in the PLC machine state table for parameters of L_pdu's. For precise definitions and coding, see Part 3C. All of the parameters are transmitted as coded in Part 3C:

DSAP = DSAP field of an L_pdu

SSAP = SSAP number and L/G bit of the SSAP field of an L_ sdu

L_pt = L_pdu_type parameter of the L_pdu_type field of an L_sdu

L_Status = Response_status parameter of the R_status field of an L_pdu

L_du = L_data_unit parameter of an L_pdu

SEQ = the sequence_number bit of the L_pdu_type field of an L_pdu. Sequence_number is type BOOLEAN and takes on the values 0 or 1.

3A.1.3 Notations used for PLC-MAC interface parameters. This section defines abbreviations used in the PLC machine state tables for parameters passed across the PLC-MAC interface. For precise definitions, see Part 4. Where noted these parameters are transmitted as coded at the PLC-MAC interface.

M_sdu = MAC service data unit parameter of a primitive at the PLC-MAC interface and in a MAC frame

SA = the source_address parameter of a primitive at the PLC-MAC interface and in a MAC frame

DA = the destination_address parameter of a primitive at the PLC-MAC interface and in a MAC frame

CC = the confirmation_class parameter of a primitive at the PLC-MAC interface

PR = the priority parameter in a primitive at the PLC_ MAC interface

M_status = the M_status parameter of a primitive at the PLC-MAC interface

3A.2 Variable definitions for the PLC machine state tables. This section defines variables used by the PLC state machines. The formats used are implementation-dependent.

3A.2.1 Global PLC variables. These variables are shared by all PLC state machines in one station:

RHIS contains the L_pdu_type, source_address, priority and sequence_number corresponding to the latest received L_pdu. Usage: Remote PLC machine.

STAT contains the R_status value of last transmitted response frame. Usage: Remote PLC machine.

3A.2.2 Local PLC variables. These variables are specific to the local PLC state machine which corresponds to each PLC invocation (see 3B.1.1).

CONTEXT = The SSAP, DSAP, remote_address, service class and priority of the request currently being processed by this PLC invocation

current_priority = the priority component of CONTEXT

current_RA = the remote_address component of CONTEXT

current_DSAP = the DSAP component of CONTEXT

TSEQ (destination station) = the latest sequence_number (0 or 1) transmitted to each supported destination station at each supported priority. The total number of TSEQ = stations_supported.

3A.2.3 Remote PLC variable. This variable is specific to the remote PLC machine which corresponds to each supported DSAP:

SEM = the semaphore which controls access to the shared_data_area associated with this machine. SEM assumes the values **busy** and **not_busy**. The shared_data_area may be accessed only when SEM = **not busy**.

3A.3 Parameter definitions for the PLC machine state tables. This section defines parameters used in the PLC state machines. The formats used are implementation-dependent.

3A.3.1 Local PLC parameters. These parameters are specific to the local PLC state machine which corresponds to each PLC invocation. These parameters are established by the station management entity when this SAP is activated as described in Parts 10A and 10B:

services for which the **INITIATOR** role has been activated.

maximum_L_sdu_length for each service activated as an **INITIATOR**.

3A.3.2 Remote PLC parameters. These parameters are specific to the remote PLC state machine which corresponds to each DSAP. These parameters are established by the station management entity when this SAP is activated as described in Parts 10A and 10B:

services for which the **RESPONDER** role of this DSAP has been activated.

maximum_L_sdu_length for each service for which the **RESPONDER** role has been activated.

shared_data_area identified when this DSAP's RDR response role was activated.

NSAP = the SSAP specified to receive L_REPLY.indications when this DSAP's RDR response component was activated

3A.4 Constants used in the PLC machine state. This section defines constants used in the PLC state machines. The formats used are implementation-dependent.

priorities_supported = the number of priorities supported in this station = **4**

stations_supported = the number of stations with which this station exchanges SDA, or RDR L_pdu's $\leq$ **256**

3A.5 Functions and Procedures — Definitions for the PLC state machines. This section defines all functions and procedures (other than the issuance of service primitives) which are performed by the PLC state machines. The service primitives are defined in Parts 2A and 4.

3A.5.1 LOCAL_STATUS?

Returns:	**Active** IF the INITIATOR role of one or more services of the SSAP underlying this machine are now activated as described in Parts 10A and 10B
Returns:	**Inactive** OTHERWISE

3A.5.2 UPDATE_CONTEXT

Returns:	None
Function:	Save SSAP, DSAP, remote_address and service_class of the request referenced on this arc in CONTEXT current_priority = **EVEN** (service_class)

3A.5.3 VALIDATE?

Returns:	**IP** IF the parameters of the current request primitive do not meet the prescription of Part 2A
Returns:	**LS** IF NOT **IP** and the INITIATOR role of the service specified by the current request primitive is not currently activated as described in Parts 10A and 10B
Returns:	**Valid** OTHERWISE

Additional Comments: The parameters of the current request primitive are referenced on any arc that invokes the VALIDATE? function.

3A.5.4 SEQUENCE (destination_address)

Returns:	Sequence_number = the TSEQ value corresponding to this MAC destination_address and priority

3A.5.5 BUILD_PDU (L_pdu_type, sequence_number, SSAP, DSAP, L_du)

Returns:	An L_pdu
Function:	Builds a command L_pdu of the format specified in Part 3C with values as supplied in the calling arguments. The sequence_number parameter is merged with the L_pdu_ type parameter to form the L_pdu_type field of the L_pdu. The C/R bit of the SSAP field is set = **0**. This L_pdu becomes a parameter of the MA.DATA.request generated on the arc which references **BUILD_PDU**.

3A.5.6 EXTRACT_CONTEXT

Returns:	The SSAP, DSAP, remote_address, and service_class now contained in CONTEXT. The order of these parameters is as shown above.
Function:	Sets CONTEXT to null

3A.5.7 NOTIFY_MGT

Returns:	None
Function:	Notifies the station management entity of a Protocol error and the conditions under which it occurred

3A.5.8 REMOTE_STATUS?

Returns:	**Active** IF the RESPONDER role in one or more services of the DSAP underlying this machine is currently activated as described in Parts 10A and 10B
Returns:	**Inactive** OTHERWISE

3A.5.9 RESOURCES?

Returns:	**Available** IF PLC resource is available and no MAC parameter indicated that MAC was without resources
Returns:	**Unavailable** OTHERWISE

3A.5.10 DUPLICATE?

Returns:	**Yes** IF the L_pdu_type, source_address, priority, and sequence_number associated with the current MA_DATA.indication all agree with the values previously saved in RHIS
Returns:	**No** OTHERWISE

3A.5.11 ACTIVATED?

Returns:	**Yes** IF the RESPONDER role for the service specified in the current MA_DATA.indication is currently activated for the DSAP underlying this machine
Returns	**No** OTHERWISE

3A.5.12 UPDATE_HISTORY

Returns:	None
Function:	Save L_pdu_type, source_address, priority and sequence_ number associated with the current MA_DATA.indication in RHIS

3A.5.13 REQUEST_DATA_AREA

Returns:	**Not_busy** IF SEM = **not_busy**
Returns:	**Busy** OTHERWISE
Function:	Set SEM = **busy** IF **not_busy** is returned

3A.5.14 RELEASE_DATA_AREA

Returns:	None
Function:	Set SEM = **not_busy**

3A.5.15 UPDATE_DATA_AREA

Returns:	None
Function:	The L_sdu of the current L_REPLY_UPDATE.request replaces the contents of the shared_data_area associated with this machine

3A.5.16 RESPONSE_TYPE (L_pdu_type)

Returns:	**SDAR** IF L_pdu_type = SDA
	RDRR IF L_pdu_type = RDR
Returns:	Error indication OTHERWISE

3A.5.17 BUILD_RPDU (L_pdu_type, sequence_number, SSAP, DSAP, R_status, L_ du)

Returns:	An L_pdu
Function:	Builds a response L_pdu of the format specified in Part 3C with values as supplied in the calling arguments. The C/R bit of the SSAP field is set = **1**. This L_pdu becomes a parameter of the MA.DATA.request generated on the arc which references **BUILD_RPDU**.

3A.5.18 EVEN (service_class)

Returns:	**0** IF service_class = 0 or 1
	2 IF service_class = 2 or 3
	4 IF service_class = 4 or 5
	6 IF service_class = 6 or 7

3A.5.19 UPDATE_SEQ (destination_address)

Returns:	None
Function:	Complements TSEQ (destination_address)

3A.6 Mandatory features

3A.6.1 Validity of response frames. The remote PLC entity shall provide a valid response for each SDA and RDR L_pdu received. Specifically the UN status must be returned in all cases where resources are not available. Updates of the shared_data_areas shall only influence responses to RDR type L_pdus.

3A.6.2 PLC_Station _ Delay. The remote PLC entity shall provide a valid response L_pdu when it receives a MA_DATA.indication containing an SDA and RDR L_pdu. **This response shall be provided in a timely manner so that the remote station complies with the PLC_station_delay requirements of 5B.1.8**.

3A.6.3 Mandatory state machines and features

3A.6.3.1 Local machine. The local machine including all of its areas and features is mandatory for **INITIATOR** stations. The local machine is not required in **RESPONDER** stations. The minimum required attributes of the local PLC entity are given in the table below.

Attribute	Minimum INITIATOR Requirement
Individual Locally Administered SSAPS	4
Globally Administered SSAPS	01110001 01000000 11000000 01110000 01110010
Priorities supported	4
Minimum, maximum L_sdu_length supported	512 octets

3A.6.3.2 Remote machine. The remote machine is mandatory for all stations. The arcs of this machine and related machine features are categorized as Mandatory = M or – for Not required for **INITIATOR** and **RESPONDER** machines in the table below. Any variable or function referenced by a mandatory arc is itself mandatory.

Arc	INITIATOR	RESPONDER
1, 3, 4, 6, 7, 8, 9, 10 and 15	M	M
2, 11, 12, 13	–	M

The minimum required attributes of the remote PLC entity for INITIATOR and RESPONDER stations are given in the table below.

Attribute	Minimum INITIATOR Requirement	Minimum RESPONDER Requirement
Individual Locally Administered DSAPS	4	1
Group Locally Assigned DSAPS	0	0
Shared Data Areas	0	1
Global DSAPS	01110001 01000000 11000000 01110000 01110010	 01000000 01110000
Priorities supported	4	4
Minimum, maximum L_sdu_length supported	512 octets	16 octets

3A.6.4 Labeling. The vendor shall label each station to show which PLC options are supported and the supported value of each PLC attribute.

3B. PLC machine formal description

3B.1 Overview. The PLC sublayer implementation in each station with Initiator functions contains one local machine for each invocation supported by this station's PLC implementation. An invocation is defined as a specific combination of one SSAP and one priority. The local state machine handles all requests from and confirmations to the local user at the priority and SSAP corresponding to this invocation.

The PLC sublayer implementation in each station with Responder functions contains one remote machine for each DSAP supported by this station's PLC implementation. This remote state machine handles all indications to the remote user at this DSAP and manages the shared_data_area associated with this DSAP. These indications usually arise as a result of frames received over the local area network.

Each state machine describes the set of operations performed to support one of these PLC invocations or DSAPs. It is defined using state machine descriptive techniques. These state machines do not specify particular implementation techniques; rather they are intended to describe the external characteristics of the PLC entity as perceived by the corresponding PLC entity in another station or by a higher layer, i.e., the user, in the same station.

3B.1.1 State machine invocations. Each machine is shown for the support of a single invocation. A separate local state machine is invoked for each priority/SSAP combination that is supported by this station's implementation of the PLC sublayer. A separate remote machine is invoked for each DSAP that is supported by this station implementation of the PLC sublayer.

Stations that are members of a single network need not support the same number of PLC invocations.

3B.1.2 State diagrams. Figures 3-2 and 3-3 diagram the local and remote state machines, respectively. These state transitions are detailed in Tables 3-1 and 3-2, which are constructed according to the techniques given in 3B.2.

3B.2 Techniques used in the PLC machine state descriptions. This section provides guidance in interpreting the PLC machine state tables.

Tables 3-1 and 3-2 display the state transitions for the local and remote PLC machines. Each includes columns for the current state, the event which causes state transition, any action(s) taken before the transition, and the next state. This combination of a current state, an event causing a transition, some actions, and the next state is known as an arc. These tables define and number all valid arcs.

The following points apply to the interpretation of the state tables:

1) There is no implied ordering to the arcs. Also arcs are mutually exclusive. Thus the first event that is satisfied causes the corresponding arc, and no other arc from the current state, to be executed.

2) An arc may terminate in the same or a different state.

3) Events which are not listed as valid inputs to the current state shall not cause state transitions.

4) Actions specified for an arc are executed in the order that they appear in the state table. No other actions are taken on the transition.

5) Functions, procedures, variables, and constants defined in Part 3A are in these tables.

6) Primitives defined in Parts 2A, 4 and 10B are referenced.

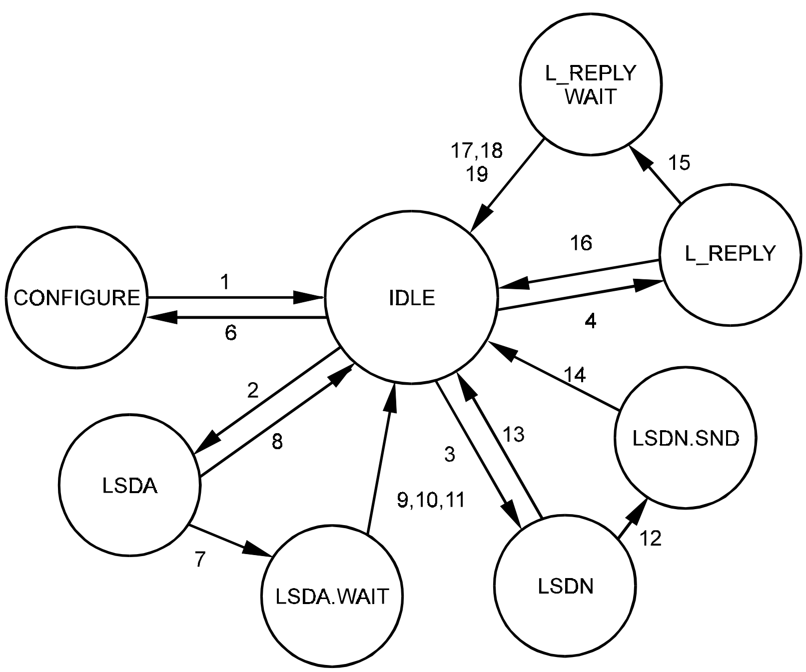


Figure 3-2: PLC local machine state diagram

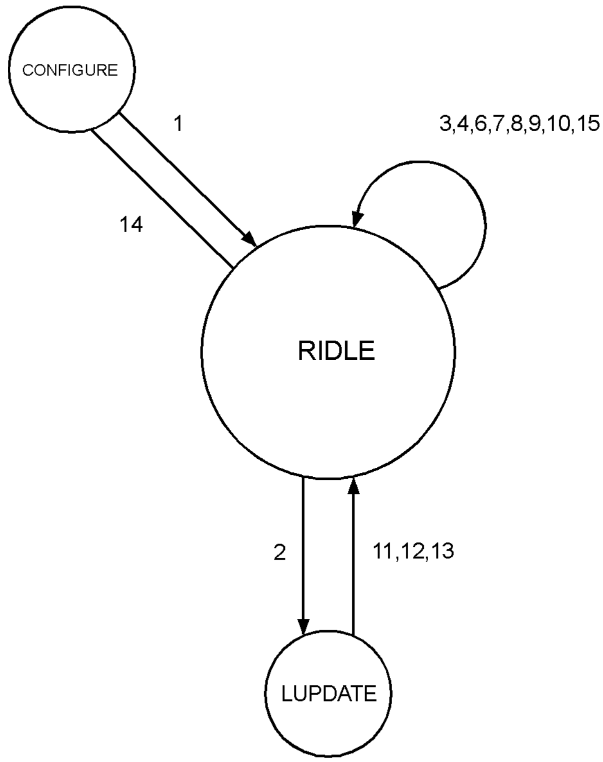


Figure 3-3: PLC remote machine state diagram

7) By convention if one and only one primitive at a given interface is mentioned in the definition of an arc, then all parameters related to that interface are parameters of that primitive. If none or more than one primitive at an interface are mentioned in the definition of an arc, then each parameter is explicitly related to the appropriate primitive.

8) When request and confirmation primitives at the same interface occur in the same arc, the confirmation parameters are those of the request unless otherwise stated.

9) The parameters listed for a primitive are the only parameters whose value are germane in deciding which (if any) arc to take.

10) Activation and configuration of LSAP's are described in Parts 10A and 10B and Appendix 10-B.

11) Error recovery is for further study.

3B.3 State transition table for local PLC state machine. Table 3-1 describes the local (originating) PLC state machine. The local machine handles service requests (except Update requests) originating from the local PROWAY user and confirmation to that local user.

Table 3-1: State transition table for the local PLC state machine

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
CONFIGURE	LOCAL_STATUS?=Active	None	IDLE	1
IDLE	L_DATA_ACK.request	UPDATE_CONTEXT	LSDA	2
IDLE	L_DATA.request	UPDATE_CONTEXT	LSDN	3
IDLE	L_REPLY.request	UPDATE_CONTEXT	L_REPLY	4
IDLE	LOCAL_STATUS?=Inactive	None	CONFIGURE	6
LSDA	VALIDATE?=Valid	BUILD_PDU (L_pt:=SDA, SEQ:=SEQUENCE (RA), SSAP:=SSAP, DSAP:=DSAP, L_du:=L_sdu) MA_DATA.request (DA:=RA, M_sdu:=L_pdu, PR:=current_priority, CC:=RR)	LSDA.WAIT	7
LSDA	VALIDATE?≠Valid	L_DATA_ack.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC),	IDLE	8
LSDA.WAIT	MA_DATA.confirm (M_status≠OK)	L_DATA_ack.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC), L_sdu:=null, L_status:=M_status	IDLE	9
LSDA.WAIT	MA_DATA.indication (SA=current_RA AND SSAP=current_DSAP AND L_pt=SDAR) AND MA_DATA.confirm (M_status=OK))	L_DATA_ack.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_sdu:=null, L_status) UPDATE_SEQ (current_RA)	IDLE	10

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
LSDA.WAIT	(MA_DATA.indication (SA≠current_RA OR SSAP≠current_DSAP OR L_pt≠SDAR) or L_du≠null))	NOTIFY_MGT.	IDLE	11
LSDN	VALIDATE?=Valid	BUILD_PDU (L_pt:SDN, SEQ:=0 SSAP:=DSAP, L=du:=L_sdu) MA_DATA.request (DA:=RA, M_sdu:=L_pdu, CC:=RQ, PR:=current_ priority)	LSDN.SND	12
LSDN	VALIDATE?≠Valid	L_DATA.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_STATUS:=VALIDATE	IDLE	13
LSDN.SND	MA_DATA.confirm	L_DATA.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_status:=M_status)	IDLE	14
LREPLY	VALIDATE?=Valid	BUILD_PDU (L_pt:=RDR, SEQ:SEQUENCE(RA) SSAP:=SSAP DSAP:=DSAP L_du:=null) MA_DATA. request (DA:=RA, M_sdu:=L_pdu, PR:=current_ priority, CC:=RR)	LREPLY. WAIT	15

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
LREPLY	VALIDATE?≠Valid	L_REPLY.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_sdu=null, L_status:=VALIDATE)	IDLE	16
LREPLY.WAIT	MA_DATA.confirm (M_status≠OK)	L_REPLY.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_sdu:=null, L_status:=M_status)	IDLE	17
LREPLY.WAIT	(MA_DATA.indication (SA=current_RA AND SSAP=current_DSAP AND L_pt=RDRR) AND MA_DATA.confirm (M_status=OK))	L_REPLY.confirm (EXTRACT_CONTEXT !SSAP, !DSAP, !RA, !SC, L_sdu, L_status), UPDATE_SEQ (current_RA)	IDLE	18
LREPLY.WAIT	(MA_DATA.indication (SA≠current_RA OR SSAP≠current_DSAP OR L_PT≠ RDRR))	NOTIFY_MGT	IDLE	19

The following arcs describe conditions that are implied in the state description of Reference 3: 2, 3, 4, 5, 8, 11, 13, 14, 16, 19.

3B.4 State transition table for the remote PLC state machine. Table 3-2 describes the remote PLC state machine which handles service indications originating from the MAC layer and manages shared_data_areas.

Table 3-2: State transition table for the remote PLC state machine

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
CONFIGURE	REMOTE_STATUS?=Active	None	RIDLE	1
RIDLE	REMOTE_STATUS?=Inactive	None	CONFIG- URE	14
RIDLE	L_REPLY_UPDATE.request	None	LUPDATE	2
RIDLE	(MA_DATA.indication (L_pt=SDN) AND RESOURCES?=Available AND ACTIVATED?=Yes	L_DATA.indication (SSAP:=SSAP, DSAP:=DSAP, LA:=SA, RA:=DA, SC:=PR, L_sdu=L_du)	RIDLE	3
RIDLE	(MA_DATA.indication (L_pt=SDA), AND L_du≠null AND RESOURCES?=Available AND ACTIVATED?=Yes AND DUPLICATE?=No	BUILD_RPDU (L_pt:=SDAR, SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP R_status:=OK, L_du:=null) MA_DATA.request (M_sdu:=L_pdu, DA:=SA, CC:=RS, PR:=PR) L_DATA_ACK.indication (SSAP:=SSAP, DSAP:=DSAP, LA:=SA, RA:=DA, L_sdu:=L_du) UPDATE_HISTORY STAT:=OK	RIDLE	4

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
RIDLE	(MA_DATA.indication (L_pt=SDA) AND DUPLICATE? =YES AND ACTIVATED?=Yes	BUILD_RPDU (L_pt:=SDAR, SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP, R_status:=STAT, L_du:=null) MA_DATA.request (M_sdu:=L_pdu, DA:=SA, CC:=RS, PR:=PR)	RIDLE	6
RIDLE	(MA_DATA.indication (L_pt=RDRR) AND REQUEST_DATA_AREA= Not Busy AND ACTIVATED?=Yes)	BUILD_RPDU (L_Pt:=RDRR< SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP, R_status:=OK, L_DU:=shared_data_ area contents) MA_DATA.request (M_sdu:=L_pdu, DA:=SA< CC:=RS, PR:=PR) L_REPLY.indication (SSAP:=NSAP, DSAP:=DSAP, LA:=SA, RA:=DA, SC:=PR, L_sdu:=null) RELEASE_DATA_AREA UPDATE_HISTORY STAT:=OK	RIDLE	7

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
RIDLE	(MA_DATA.indication (L_pt=RDR) AND (REQUEST_DATA_AREA =Busy OR RESOURCES?= Unavailable AND ACTIVATED?=Yes)	BUILD_RPDU (L_pt:=RDRR, SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP, L_status:=UN, L_du:=null) MA_DATA.request (M_sdu:=L_pdu, DA:=SA, CC:=RS, PR:=PR) UPDATE_HISTORY STAT:=UN	RIDLE	8
RIDLE	(MA_DATA.indication (L_pt=SDA AND L_du= null AND RESOURCES?= Available AND ACTIVATED?=Yes AND DUPLICATE?=No	BUILD_RPDU (L_pt:=SDAR, SEQ:=NOT (SEQ), SSAP:=DSAP, DSAP:=SSAP, R_status:=OK, L_du:=null) MA_DATA.request (M_sdu:=L_pdu, DA:=SA, CC:=RS, PR:=PR) UPDATE_HISTORY STAT:=OK	RIDLE	15

CURRENT STATE	EVENT	ACTION	NEXT STATE	ARC #
RIDLE	(MA_DAT.indication (CC=RR) AND ACTIVATED?=No	BUILD_RPDU (L_pt:=REPONSE_TYPE (L_pdu_type), SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP, R_status:=RS, L_du:=null) MA_DATA.request (M_sdu:L_pdu, DA:=SA, CC:=RS, PR:=PR) UPDATE_HISTORY STAT:=RS	RIDLE	9
RIDLE	(MA_DATA.indication (L_pt=SDA) AND DUPLICATION?=No AND ACTIVATED?=Yes RESOURCES?=Unavailable)	BUILD_RPDU (L_pt:=SDAR, SEQ:=NOT(SEQ), SSAP:=DSAP, DSAP:=SSAP, R_status:=UN, L_du:=null MA_DATA.request (M_sdu:=L_pdu, DA:=SA, CC:=RS, PR:=PR) STAT:=UN	RIDLE	10
LUPDATE	(RVALIDATE?=Valid AND REQUEST_DATA_AREA= Not Busy)	UPDATE_DATE_AREA L_REPLY_UPDATE.confirm (DSAP:=DSAP, L_status:=OK) RELEASE_DATE_AREA	RIDLE	11
LUPDATE	(RVALIDATE?≠Valid)	L_REPLY_UPDATE.confirm (DSAP:=DSAP L_status:=RVALDIATE)	RIDLE	12
LUPDATE	VALIDATE?=Valid AND REQUEST_DATA_AREA= Busy)	L_REPLY_UPDATE.confirm (DSAP:_DSAP, L_status:=UN	RIDLE	13

The following arcs describe conditions that are implied in the state descriptions of References 3: 1,11,12,13,14.

3C. PLC protocol data unit format

3C.1 Function of the PLC_protocol_data_unit. The PLC_protocol _ data _unit (L_pdu) is used to transfer data, commands or status information between cooperating PLC entities.

The **send_data_with_no_acknowledge (SDN) L_pdu** is used by the local PLC to convey an L_du to one or more remote stations without requiring an acknowledge. The SDN L_pdu is invoked by the L_DATA.request primitive.

The **send_data_with_acknowledge (SDA) L_pdu** is used by the local PLC to convey an L_pdu to one remote station and to request an acknowledge from that remote station. The SDA L_pdu is invoked by the L_DATA_ACK.request primitive.

The **request_data_with_response (RDR) L_pdu** is used by the local PLC to pass a request for data to a remote station. The RDR L_pdu is invoked by the L_REPLY.request primitive.

3C.2 Structure of the PLC_protocol _ data _ unit. Each PLC _ protocol_data_unit (L_pdu) must contain a 3 or 4 octet PLC header. In addition the L_pdu may contain a Link_data_unit (L_du), i.e., information field.

The format of the PLC L_pdu is:

PLC_header	L_du
3 or 4 octets	0 to 1000 octets

First field delivered to or received from the MAC sublayer.

The structure of the L_pdu depends on the service that is requested. The structure of the L_pdu for each PROWAY service is given in Table 3-3.

.

Table 3-3: PLC protocol data unit structure

L_pdu_type	L_pdu_type Coding	R_status Present	Command/ Response	L_du Present
SND=Send data with no ack.	11000000	No	Command	Yes
SDA=Send data with ack.	1110011S	No	Command	Yes
RDR=Request data with resp.	1110111S	No	Command	No
SDAR=SDA Response	1110011S	Yes	Response	No
RDRR	1110111S	Yes	Response	Yes
	1st bit delivered_to or received_from MAC			

S=Sequence_number

NOTE: The above bit patterns are equivalent to those for the type 1 and type 3 services of Reference 3 of Appendix I-A.

3C.2.1 PLC header composition. The PLC header specifies the PLC L_pdu_type of this L_pdu and provides additional information that depends on the L_pdu_type. The PLC header consists of

a DSAP field

a SSAP field

a L_pdu_type field

a R_status field for response L_pdu_types as shown in Table 3-3.

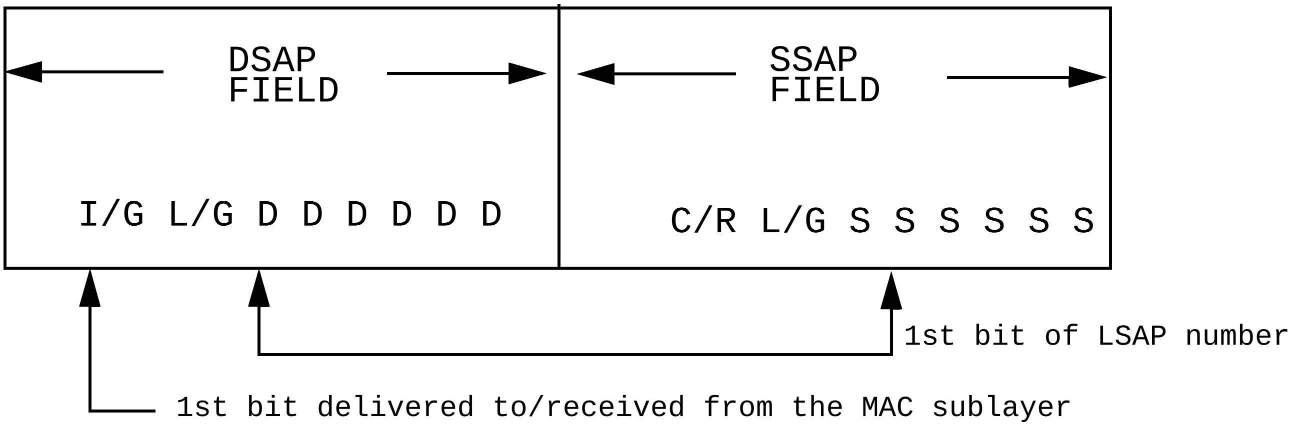
The format of the PLC_header is as shown below.

lst octet delivered to or received from MAC

DSAP Field	SSAP Field	L_pdu type	R_status
1 octet	1 octet	1 octet	0 or 1 octet

3C.2.2 Service_access points. Each PLC L_pdu shall contain two link service_access point (LSAP) fields: the destination service_access point (DSAP) field and the source service_access point (SSAP) field. The DSAP shall identify the one or more service_access points for which the PLC L_du is intended. The SSAP shall identify the specific access point from which the L_du was initiated.

3C.2.2.1 LSAP representation. The LSAP representation shall be as shown below.




```
I/G = 0  INDIVIDUAL DSAP
I/G = 1  GROUP DSAP
C/R = 0  COMMAND PDU
C/R = 1  RESPONSE PDU
L/G = 0  LOCALLY ASSIGNED LSAP
L/G = 1  LSAP IS RESERVED FOR DEFINITIONS BY STANDARDS BODIES

LSAP_NUMBER RANGE = 0-63
```

3C.2.2.2 LSAP field composition

3C.2.2.2.1 Each LSAP field shall contain one octet.

3C.2.2.2.2 Each LSAP field shall contain 6 bits of the actual LSAP number.

3C.2.2.2.3 The I/G bit shall be the first bit delivered to the MAC of the DSAP field. If this bit is "0", it shall indicate that the address is an individual DSAP address. If this bit is "1", it shall indicate that the address is a group DSAP address that identifies none, one or more, or all of the service_access points that are serviced by the PLC entity. Group DSAP addresses are allowed only if the L_pdu_type is SDN.

3C.2.2.2.4 The C/R bit shall be the first bit delivered to the MAC of the SSAP field. If this is "0", it shall indicate that the L_pdu is a command. If this bit is "1", it shall indicate that the L_pdu is a response as shown in Table 3-3.

3C.2.2.3 LSAP address usage. An individual LSAP shall be usable as both an SSAP and a DSAP address. A group or the global LSAP shall be used only as a DSAP address and only in conjunction with SDN L_pdu's.

3C.2.2.4 Global administered LSAP assignments

3C.2.2.4.1 Global DSAP. DSAP fields = 11111111 (all "1s") is the Global DSAP. This DSAP designates an SDN L_pdu as destined for all DSAPs actively being serviced by the underlying MAC entity.

3C.2.2.4.2 PROWAY application LSAP. The LSAP = 01110010 is reserved for higher layer PROWAY applications.

3C.2.2.4.3 Station management LSAPs. LSAP field = 01110000 is designated as the individual LSAP for this station's station management entity.

3C.2.2.4.4 Active_station_list LSAP. LSAP field = 01110001 is designated as the SSAP and DSAP used to exchange Mgt_pdu's related to maintenance of the active_station_list as defined in Part 10E.

3C.2.2.5 Locally administered LSAP assignments. Recommended values of locally administered individual and group LSAP numbers in PROWAY systems are for further study.

3C.2.3 Specification of shared data areas. For the RDR service an individual DSAP also designates a particular shared_data_area. This shared_data_area is updated by L_UPDATE.request primitives designating this LSAP.

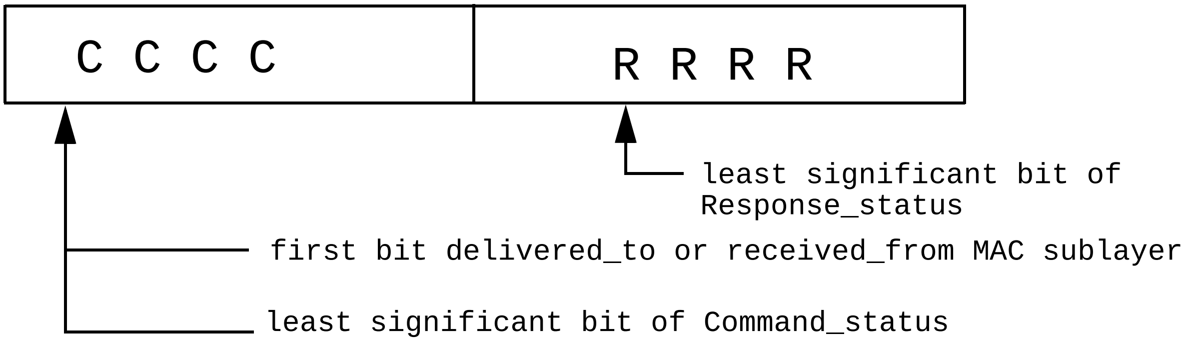
3C.2.4 Implied SAP. The user identified by the user notification LSAP will receive notifications of changes in ring membership as defined in Part 10E and changes in station status as defined in Part 10A.

3C.2.5 L_PDU_TYPE definitions. The L_pdu_type specifies the underlying service to be provided by the PLC layer. It also specifies the composition of the L_pdu as shown in Table 3-3 and carries sequence_number information to assist in detection of duplicate frames.

3C.2.5.1 Sequence_numbers. SDA, RDR, SDAR and RDRR L_pdu's have the sequence_number bit set as specified in Part 3B. The sequence_number bit is reserved and set equal zero for SDN L_pdu's.

3C.2.5.2 R_status definition. R_status conveys the disposition of the immediately preceding command L_pdu by the remote MAC and/or PLC entities.

The format of the R_status field is shown below.



T = Temporary error. A future retry of the associated transmission is likely to succeed.

P = Permanent error. Management intervention may be required before a retry of the associated transmission is likely to succeed.

3C.3 L_Data_Unit

3C.3.1 Data unit size. The L_du shall consist of any integral number (including zero) of octets. The maximum length of an L_du in a PROWAY system is 1000 octets.

Command_Status			
Value	Code	Meaning	T/P
0	OK	Command LSDU accepted	_
1	RS	Request for an unimplemented or unactivated service or receipt of an RDR pdu containing a non-null data unit at remote DSAP: no action taken	P
5	UE	User-PLC interface error	P
6	PE	Protocol error	P
7	IP	Permanent implementation dependent error	P
9	UN	Resources not available to remote PLC or MAC entity: no action taken	T
15	IT	Temporary implementation dependent error	T
Others		Reserved	_

Response_Status			
Value	Code	Meaning	T/P
0	OK	Response LSDU present	—
1	RS	Request for an unimplemented service: no action taken	P
3	NE	Response LSDU never submitted to remote PLC	P
4	NR	Response LSDU not requested	—
5	UE	User-PLC interface error	P
6	PE	Protocol error	P
7	IP	Permanent implementation dependent error	P
9	UN	Resources not available to remote PLC or MAC entity: no action taken	T
15	IT	Temporary implementation dependent error	T
Others		Reserved	—

3C.3.2 Bit order. The information field shall be delivered to the source MAC sublayer in the same bit order as received from the source user of PLC. The information field shall be delivered to the destination user of PLC in the same bit order as received from the destination MAC sublayer.

3C.4 Invalid L_pdu. An invalid L_pdu shall be defined as one which meets at least one of the following conditions:

1) It is identified as such by the Physical layer or the Medium Access Control (MAC) sublayer.
2) It is not an integral number of octets in length.
3) It does not contain a properly formatted PLC header and an L_du corresponding to Table 3-3. Note that the L_du length may be 0 octets in an L_pdu that contains an L_du.
4) Its length is less than 3 octets.

Invalid L_pdu's shall be ignored by the destination PLC entity.

Appendix 3-A Prevention of loss and duplication for the PROWAY confirmed data transfer services

3-A.1 Introduction. This appendix is explanatory in nature. It is not a requirement of this standard.

The PROWAY **confirmed services** (SDA and RDR) are designed to prevent loss and duplication of messages. This appendix provides a general overview of how this protection is achieved. It first describes the underlying characteristics of the confirmed services. This is followed by a detailed examination of loss and duplication scenarios. The analysis shows that loss prevention adds a source of duplication, which must be accommodated by a sequence_number generated by the local (originating) PLC entity and checked by the remote (responding) PLC entity.

3-A.2 Overview of the PROWAY confirmed services. The PROWAY confirmed services are provided by the PLC sublayer in conjunction with the token bus MAC sublayer. When the local user issues a request for an SDA or RDR service, the local PLC queues a request to the MAC layer at the specified priority. The sequence_number bit of the PLC L_pdu_type field is toggled, as described below, in order to prevent frame duplication. The local MAC composes the required request_with_response frame.

When the local MAC obtains the token, it sends the request_with_response frame and waits for an immediate response. It does not transmit any other frames nor pass the token while waiting. If a timer expires, the local MAC will resend the original request_with_response frame. This is repeated until a response is received or the retry counter exceeds the maximum_retry_limit, which had been previously set by station management.

When the remote MAC receives any frame, it passes the received frame up to the remote PLC. If the frame is of L_pdu_type = SDA or RDR, it is the responsibility of the remote PLC to make the determination as to whether the data unit is a duplicate. If the frame is not a duplicate or is for the SDN service, the data unit contained in the frame is passed to the remote user.

For all request_with_response frames (L_pdu_type = SDA or RDR) the remote PLC immediately generates the appropriate response (acknowledgment or reply) and directs the remote MAC to send this response to the source of the request_with_response frame. The sequence_number bit of the request_with_response frame is returned in the response frame to maintain compatibility with other standards.

When the originating MAC receives the response frame or the retry counter is exhausted, it notifies the local PLC of completion of the originating request.

3-A.3 Sources of frame loss. Two potential sources of frame loss in the confirmed services are:

First. Loss of frames due to line errors which cause the Request frame or its associated response frame to be lost.

Second. Loss of a request_with_response frame because the remote station incorrectly decided that frame is a repeat, discards it and repeats its previous response. This may cause the originating station to incorrectly assume that the request_with_response frame was properly received.

3-A.4 Protection against loss and creation of duplicate frames. The prevention mechanism for the first type of loss is the MAC level retry mechanism at the originating station. If a response is not detected before the timeout expires, the frame is retransmitted and the timer is restarted. This is repeated until the retry counter exceeds the limit established by station management or a response is heard. If the retry counter is exhausted, local user is notified that the link has failed. In this case the only unknown is whether the remote station correctly received any of the frames transmitted.

If the remote station has received a frame that the local station thinks was lost, a potential for duplication on the next token round exists. The level of protection against duplication is R, the maximum_retry_limit, since R request_with_response or response frames must be lost to create the above condition.

The retry mechanism prevents loss except in the case where the retry counter is exhausted. In this case, the local user must send an SDA frame with a null data unit to regain synchronization with the remote user.

There is a failure scenario with such a scheme if the response delays are not bounded. If a frame is sent and the timeout expires before the response is returned, a second copy of the frame is sent, which is also confirmed. When the first acknowledge is received, the originator sends a new frame, which may get lost. However, the response from the retransmission of the first frame may be received and incorrectly associated with the second data frame. In the PROWAY specification, the reply must be sent by the remote station within 2 octet times. So long as the retry timeout is set longer than 3 slot times, the scenario described above will not occur.

3-A.5 Protection against duplication. Duplication occurs at the remote station if it cannot differentiate between a retry of a previously received frame and a new frame. The PROWAY specification uses a single sequence_number bit to solve the duplication problem.

The remote PLC makes the duplicate determination based on the last received source_address, SSAP, priority, and the state of the sequence_number bit. If all fields match and the previous frame was accepted, then the frame will be rejected as a duplicate and the previous status is returned as the response. Otherwise the frame is treated as a new frame. Note that the response to this frame may be generated immediately by PLC prior to checking the address fields if buffer space is available to store the incoming frame. This alleviates the processing load required before generating the response.

Each time the local PLC entity queues a request_with_response frame to the remote station at a given priority, it will have to toggle the sequence_number bit of the PLC L_pdu_type field to allow the remote PLC to differentiate a new request_with_response frame from a retry.

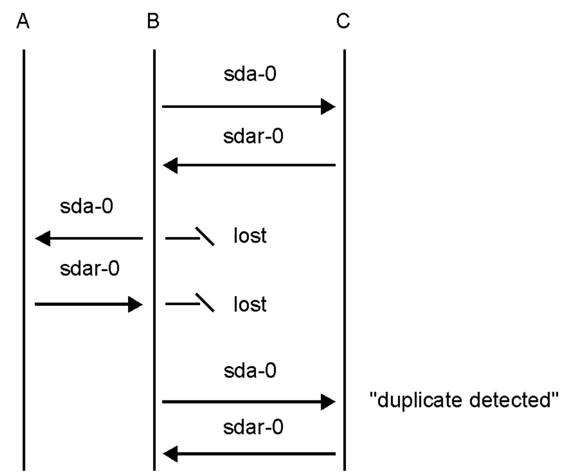
The local PLC must maintain a history and state of the sequence_number bit for the previous request_with_response frame queued to each destination station for each priority.

3-A.6 Operation with non_PROWAY MAC layers. The confirmed service depends on the token bus request_with_response procedure which prevents independent traffic from intervening between the request frame and its response. A confirmed connectionless service can prevent loss of frames using non-PROWAY MACs by implementing Link level retry procedures. Prevention of frame duplication using a non-PROWAY MAC requires maintenance of state information for each pair of communicating LSAPS.

3-A.7 An example of erroneous detection of duplication when sequence information is not used. Another loss occurs if the originating station queues two different request_with_response frames with identical source address, destination address and priority while the destination station sees nothing between them. This can be caused by the destination station missing the intervening frames or the source failing to insert frames.

A station may send two request_with_response frames with the same source address, destination address, and frame control field sequentially.

If toggling of the sequence bit was not used, duplicating could occur (due to lost frames at destination station). As shown in the following example, if station C misses the request_with_response and response between B and A, it will assume that the next message it receives is a duplicate. Note that the transmissions between B and A could be a pair of token passes or a request_with_response.



4 PLC-MAC interface and service specification

4.1 Scope and field of application. This section specifies the services provided to the PROWAY Link Control (PLC) sublayer at the boundary between the Link Control sublayer and the Medium Access sublayer of the Data Link (HIGHWAY) layer of the ISO reference model. This standard specifies these services in an abstract way. It does not specify or constrain the implementation entities and interfaces within a computer system. The relationship of this section to other sections of this standard and to LAN specifications is illustrated in Figure 4-1.

4.2 Overview of the PLC-MAC data transfer service

4.2.1 General description of services provided. This section describes informally the MAC_DATA transfer service provided to the PLC sublayer by the MAC sublayer. This service supports data transfer between peer PLC entities. It provides the means by which PLC entities can exchange Link_protocol_data_units (L_pdu). The data transfer can be open-loop, closed-loop, point-to-point or multi-point.

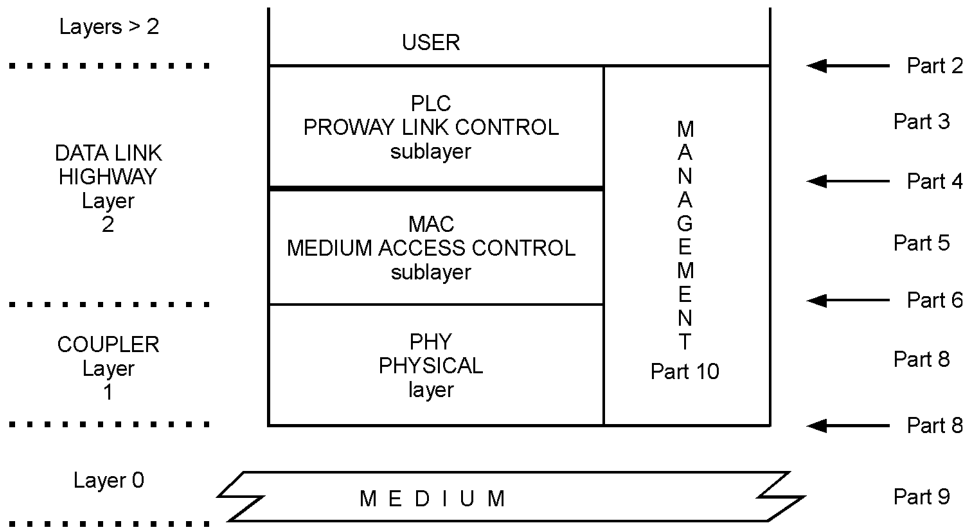


Figure 4-1: Relationship to LAN model

4.2.2 Model used for the service specification. The model and descriptive method are provided in Reference 1, Appendix I-A.

4.2.3 Notation. In this part only, stations, MAC entities and PLC entities are frequently identified by their relationship to the associated frame; i.e., as source or destination. This notation avoids the ambiguities that arise when both the local (originating) and remote (receiving) PLC entities submit a MA_DATA.request for the same service, e.g., SDA, RDR, RSR.

4.2.4 Overview of interactions. The primitives associated with the MA_DATA data transfer service are

MA_DATA.request

MA_DATA.indication

MA_DATA.confirm

MA_DATA.request primitive is passed from the source PLC entity to the source MAC entity to request that an M_sdu be sent to one, a group of, or all destination stations.

MA_DATA.indication primitive is passed from the destination MAC entity to the destination PLC entity to indicate the arrival of an M_sdu.

MA_DATA.confirm primitive is passed from the source MAC entity to the source PLC entity to convey the results of the corresponding MA_DATA, request primitive.

4.2.5 Mandatory features. The MA_DATA service and all its primitives are mandatory and are required in all implementations.

4.3 Detailed interactions with the PLC entity.

This section describes in detail the primitives and parameters associated with the MA_DATA data transfer service. Note that the service is specified in an abstract sense. The parameters specify the information that must be available to the sending entity. A specific implementation is not constrained in the method of making this information available. For example, the M_sdu parameter associated with some of the data transfer service primitives may be provided by actually passing the MAC_service_data_unit, by passing a descriptor, or by other means. The values of some selection parameters may also be implied by an implementation.

4.3.1 MA_DATA.request

4.3.1.1 Function. This primitive is the service request primitive for the MA_DATA data transfer service.

4.3.1.2 Semantics. The primitive shall provide parameters as follows:

```
MA_DATA_request
  (destination_address,
   M_sdu,
   confirmation_class,
   priority)
```

The **destination_address parameter** specifies the MAC entity address of the destination station or stations, as defined in Part 5D. The destination address may be either an individual address, a group address, or the broadcast address.

The **M_sdu parameter** specifies the MAC_service_data_unit to be transmitted by the source MAC entity for the source PLC entity.

The **confirmation_class parameter** specifies whether a response is required from the remote MAC entity and identifies such a response. The possible values of the confirmation class parameter are:

RQ = request_with_no_response
RR = request_with_response
RS = response

The **priority parameter** specifies the desired MAC priority, as defined in Part 5D.

4.3.1.3 When generated. This primitive is passed from a source PLC entity to the source MAC entity to request that the MAC entity compose and transmit a frame with the specified priority and confirmation class on the local area network.

4.3.1.4 Effect of receipt. Receipt of this primitive causes the source MAC entity to attempt to compose and transmit the specified frame.

4.3.1.5 Additional comments. Group and broadcast destination_addresses may be used only with confirmation_class = RQ.

A value of request_with_response for the confirmation_class parameter indicates that the next MA_DATA.indication should have a confirmation_class of response, in which case that next MA_DATA.indication shall be associated with this MA_DATA.request.

A value of response for the confirmation_class parameter indicates that the immediately prior MA_DATA.indication shall have had a confirmation_class of request_with_response.

4.3.2 MA_DATA.indication

4.3.2.1 Function. This primitive is the service indication primitive for the MA_DATA data transfer service.

4.3.2.2 Semantics. The primitive shall provide parameters as follows:

MA_DATA.indication
(destination_address,
source_address,
M_sdu,
confirmation_class,
priority)

The **destination_address and source_address parameters** specify the DA and SA fields, as defined in Part 5D, of a frame received by the destination MAC entity.

The **M_sdu parameter** specifies the MAC_service_data_unit received by the destination MAC sublayer entity.

The **confirmation_class and priority parameters** specify the quality of service of the frame received by the destination MAC entity. The semantics are identical to 4.3.1.2.

4.3.2.3 When generated. This primitive is passed from the destination MAC entity to the destination PLC entity to indicate the arrival of a frame from the PHY entity of the destination station. Such frames are reported only when they are free of detected errors and their (individual, group, or broadcast) destination address designates the destination MAC entity.

4.3.2.4 Effect of receipt. The effect of receipt of this primitive by the destination PLC entity is specified in Part 3B.

4.3.2.5 Additional comments. If delivered, the contents of the M_sdu parameter are logically complete and unchanged relative to the M_sdu parameter in the associated MA_DATA.request at the source station.

NOTE: This is a guarantee of transparency.

A value of request_with_response for the confirmation_class parameter indicates that the receiving PLC entity should immediately respond with an MA_DATA.request which itself has a confirmation_ class of response.

A value of response for the confirmation_class parameter indicates that this MA_DATA.indication may be associated with a prior MA_DATA.request which itself had a confirmation_class of request_with_response, and which was issued by the same PLC entity.

4.3.3 MA_DATA.confirm

4.3.3.1 Function. This primitive is the service confirmation primitive for the MA_DATA data transfer service.

4.3.3.2 Semantics. This primitive shall provide parameters as follows:

MA_DATA.confirm
(confirmation_class,
priority,
M_status)

The **confirmation_class and priority parameters** specify the quality of service actually provided. The semantics are identical to 4.3.1.2.

The **M_status parameter** specifies the success or failure of the service associated with the corresponding MA_DATA.request. M_status may assume values as listed below.

Code	Meaning
OK	= Requested service performed {Frame transmitted for confirmation_class = RQ or RS} {Response fame received for confirmation_class = RR}
DS	= Source station disconnected from line
WD	= Source station watchdog timed out
IP	= Invalid parameter (detected at source)
TE	= No acknowledgment from remote station after specified retries (possibly due to a non-PROWAY station)

4.3.3.3 When generated. This primitive is passed from the source MAC entity to the source PLC entity to confirm the success or failure of the service associated corresponding MA_DATA.request.

Thus when confirmation_class = RQ or RS, the MA_DATA.confirm is passed immediately after the attempt to transmit the RQ or RS frame.

When confirmation_class = RR, the MA_DATA.confirm is passed:

1) immediately if the RR frame cannot be transmitted, or
2) when the requested RS frame is received, or
3) when all predefined retries have been exhausted.

4.3.3.4 Effect of receipt. The effect of receipt of this primitive by the local PLC entity is specified in Part 3B. Receipt of this primitive by the remote PLC has no effect.

4.3.3.5 Additional comments. Success indicates that the requested M_sdu has been transmitted correctly to the best of the source MAC entity's knowledge.

In the case that the corresponding MA_DATA.request had a confirmation_class parameter specifying request_with_response, the MA_DATA.confirm is associated with the MA_DATA.indication conveying the response, if any such occurred.

5 The medium access control (MAC) sublayer

Part 5 specifies the Medium Access Control sublayer of the Data Link (Highway) layer of the ISO reference model.

Part 5A provides a description of the token bus medium access control mechanism including its operational and exception recovery functions. This part is intended to assist the reader in understanding the MAC sublayer and its operation. Where statements in this part conflict with Section 5B, 5C, 5D, Reference 4, Appendix I-A, or are incomplete, the other part shall take precedence.

Part 5B, MAC Sublayer Definitions and Requirements, contains precise definitions of MAC-specific terms and mandatory aspects of the medium access mechanism which are not included in Sections 5D or 5C.

Part 5C, ACM Formal Description, describes the required behavior of the access control machine of the MAC sublayer. The state tables that specify the ACM are given in Reference 4.

Part 5D defines the required MAC frame formats. This includes all allowed frame formats and the arrangement of all frame subfields.

The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 5-1.

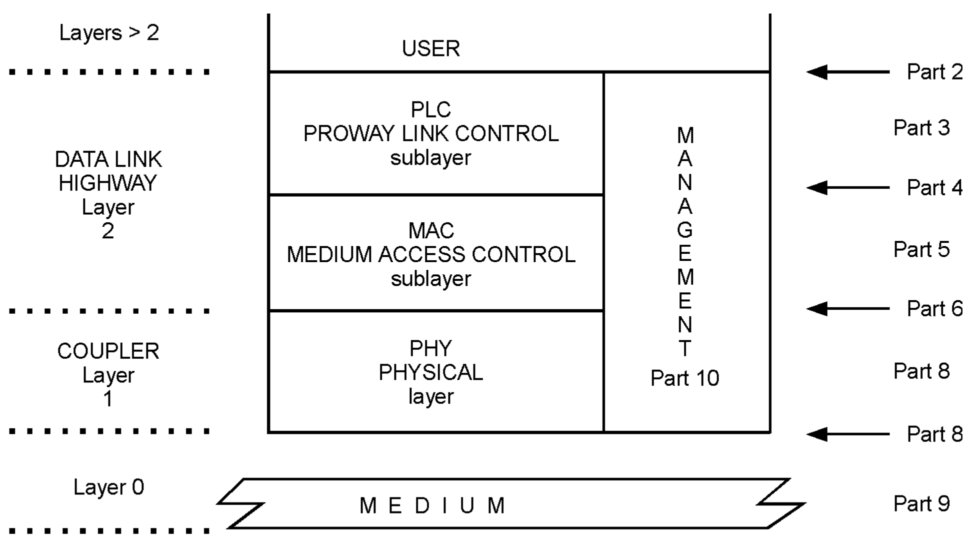


Figure 5-1: Relationship to LAN model

5A. Informal description of MAC sublayer operation

Specific responsibilities of the Medium Access Control sublayer for a broadcast medium involve managing ordered access to the medium, providing a means for admission and deletion of stations (adjustment of logical ring membership), and handling fault recovery.

The faults considered here are those caused by communications errors or station failures. These faults include the following:

1) Multiple tokens
2) Lost tokens
3) Token-pass failure
4) "Deaf" station (that is, a station with an inoperative receiver)
5) Duplicate station addresses

This medium access protocol is intended to be robust, in the sense that it should tolerate and survive multiple concurrent errors.

Some basic observations are useful in understanding the operation of tokens on a broadcast medium.

1) Stations are connected in parallel to the medium. Thus, when a station transmits, its signal is received (or "heard") by all stations on the medium. Other stations can interfere with the first station's transmission but cannot predictably alter its contents.

2) When a station transmits, it may assume that all other stations hear something (though not necessarily what was transmitted).

3) When a station receives a valid frame (properly formed and delimited and containing a correct frame check sequence), it may infer that some station transmitted the frame and, therefore, that all stations heard something.

4) When a station receives something other than a valid frame (that is, noise), it may make no inference about what the other stations on the medium might have heard.

5) Not all stations need to be involved in token passing (only those which desire to initiate transmissions).

6) Multiple tokens and lost tokens may be detected by any station. There are no special "monitor" stations that perform token recovery functions.

7) Due to spatial separation, stations cannot be guaranteed to have a common perception of the system state at any instant. (The medium access protocol described herein accounts for this.)

5A.1 Token ring steady state operation. Steady state operation (the network condition where a logical ring has been established and no error conditions are present) simply requires the sending of the token to a specific successor station as each station is finished transmitting (see Figure I-2).

Other essential and more difficult tasks are establishing the logical ring (at initialization or reestablishing it in the case of a catastrophic error) and maintaining the logical ring (allowing stations to enter and leave the logical ring without disrupting the other stations in the network.

The right to transmit, the token, passes among all stations in the logical ring. Each participating station knows the address of its predecessor (the station that transmitted the token to it), referred to as Previous Station or PS. It knows its successor (which station the token should be sent to next), referred to as Next Station or NS. It knows its own address, referred to as This Station or TS. These predecessor and successor stations are dynamically determined and maintained by the algorithms described.

The following introduce major elements and features of the token bus access protocol.

NOTE: For the purpose of description, all state machines are presumed to be instantaneous with respect to external events.

5A.1.1 Slot_time. In describing the access operations, the term slot_ time is used to refer to the maximum time any station need wait for an immediate medium access level response from another station. Slot_time is precisely defined in 5B.1.10.

The slot_time (along with the station's address and several other station management parameters) must be known to the station before it may attempt to transmit on the network. If all stations in a network are not using the same value for slot_time, the medium access protocol may not operate properly. The method of setting these parameters in each station is outside the scope of this standard.

5A.1.2 Token passing. The token (right to transmit) is passed from station to station in numerically descending, station-address order. When a station hears a token frame addressed to itself, it "has the token" and may transmit data frames.

When a station has the token it may temporarily delegate its right to transmit to another station by sending a request_with_response data frame. When a station hears a request_with_response data frame addressed to itself it must respond with a response data frame, if the request with response option is implemented. The response data frame causes the right to transmit to revert back to the station which sent the request_with_response data frame.

After each station has completed transmitting any data frames it may have and has completed other maintenance functions (described in 5A.1.3), the station passes the token to its successor by sending a "token" MAC_control frame.

After sending the token frame, the station listens to make sure that its successor hears the token frame and is active. If the sender hears a valid frame following the token, it assumes that its successor has the token and is transmitting. If the token sender does not hear a valid frame following its token pass, it must attempt to assess the state of the network.

If the token-sending station hears a noise_burst or frame with an incorrect FCS, it cannot be sure from the source address which station sent the frame. The medium access protocol treats this condition in a way which minimizes the chance of the station causing a serious error. If a noise_burst is heard, the token-sending station sets an internal indicator and continues to listen in the check_token_pass state for up to four more slot times. If nothing more is heard, the station assumes that it heard its own token that had been garbled and so repeats the token transmission. If anything is heard during the following four slot time delay, the station assumes its successor has the token.

If the token holder does not hear a valid frame after sending the token the first time, it repeats the token pass operation once, performing the same monitoring as during the first attempt.

If the successor does not transmit after a second token frame, the sender assumes that its successor has failed. The sender then sends a who_follows frame with its successor's address in the data field of the frame. All stations compare the value of the data field of a who_follows frame with the address of their predecessor (the station that normally sends them the token). The station whose predecessor is the successor of the sending station responds to the who_follows frame by sending its address in a set successor frame. The station holding the token thus establishes a new successor, bridging the failed station out of the logical ring.

If the sending station hears no response to a who_follows frame, it repeats the frame a second time. If there is still no response, the station tries another strategy to re-establish the logical ring. The station now sends a solicit_successor_2 frame with its own address as both DA and SA, asking any station in the system to respond to it. Any operational station that hears the request and needs to be part of the logical ring responds, and the logical ring is reestablished using the response window process discussed next.

If all attempts at soliciting a successor fail, the station assumes that a fault may have occurred; either all other stations have failed, all stations have left the logical ring, the medium has broken, or the station's own receiver has failed so that it cannot hear other stations who have been responding to its requests. Under such conditions the station quits attempting to maintain the logical ring. If the station has no frames to send, it listens for some indication of activity from other stations. If the station has data frames to send, it sends its remaining data frames and then repeats the token pass process. Once the station has sent its frames and still cannot locate a successor it will become silent, listening for another station's transmissions.

In summary, the token is normally passed from station to station using a short token pass frame. If a station fails to pick up the token, the sending station uses a series of recovery procedures that grow increasingly more drastic as the station repeatedly fails to find a successor station.

5A.1.3 Adding new stations to the token ring. New stations are added to the logical ring through a controlled contention process using "response windows". A response window is a controlled interval of time (equal to one slot_time) after transmission of a MAC control frame in which the station sending the frame pauses and listens for a response. If the station hears a transmission start during a response window, the station continues to listen to the transmission, even after the response window time expires, until the transmission is complete. Thus the response windows define the time interval during which a station must hear the beginning of a response from another station.

The two frame types, solicit_successor_1 and solicit_successor_2, indicate the opening of response windows for stations wishing to enter the logical ring. The solicit_successor frame specifies a range of station addresses between the frame source and destination addresses. Stations whose addresses fall within this range and who wish to enter the logical ring respond to the frame.

The sender of a solicit_successor frame transmits the frame and then waits, listening for a response in the response window following the frame. Responding stations send the frame sender their set_successor frame which requests inclusion in the logical ring. If the frame sender hears a valid request, it allows the new station to enter the logical ring by changing the address of its successor to the new station and passing its new successor the token.

In any response window there exists the possibility that more than one station will simultaneously desire logical ring entry. To minimize contention when this happens, the token pass sequence is limited by requiring that a station only request admission when a window is opened for an address range that spans its address.

There are two solicit_successor frames. Solicit_successor_1 has one response window following. Solicit_successor_2 has two response windows. Solicit_successor_1 is sent when the station's successor's address is less than the station's address. This is the normal case when the token is being passed from higher to lower addressed stations. Solicit_successor_1 allows only stations whose address is in the range between the token sender and the token destination to respond, thus limiting the possible contenders and preserving the descending order of the logical ring.

Exactly one station in the logical ring has its station's address below that of its successor, that is, the unique station having the lowest address which must send the token to the “top” of the address-ordered logical ring. When soliciting successors, this station must open two response windows, one for those stations with addresses below its own, and one for stations with addresses above its successor. The station with the lowest address sends a solicit_successor_2 frame when opening response windows. Stations having an address below the sender respond in the first response window, while stations having an address higher than the sender's successor respond in the second response window.

In any response window, when the soliciting station hears a valid set_successor frame, it has found a new successor. When multiple stations simultaneously respond, only unrecognizable noise may be heard during the response period. The soliciting station then sequences through an arbitration algorithm to identify a single responder, by sending a resolve_contention frame. The stations which had responded to the earlier solicit_successor frame and which have not yet been eliminated by the iterative resolve responders algorithm, choose a two-bit value from the station's address and listen for 0, 1, 2, or 3 slot_times as determined by that listen delay value. (This listen delay value is further described later.) If these contending stations hear anything (that is, non-silence) while listening, they eliminate themselves from the arbitration. If they hear only silence, they continue to respond to further resolve_contention requests from the soliciting stations.

NOTE: By knowing and controlling the frequency with which response windows are opened and by virtue of the finite length of the response resolution algorithm, hard bounds on the access delay can always be calculated (determinism). For 16-bit addresses, the response resolution cycle needs to be run a maximum of nine times (16/2 + 1, for 16 address bits taken two bits at a time, plus one pair of "random" bits).

5A.1.3.1 Bound on token rotation time. A maximum token rotation time for ring maintenance is established with the ring maintenance timer, similar to the rotation times established for the differing access classes of data transmissions, as described in 5A.1.5. If the token appears to rotate slower than the time established for this ring maintenance timer, a station defers the solicit successor procedure until a later token pass. When the network is less heavily loaded on the next or succeeding token pass, the station performs the ring maintenance function of soliciting new successors.

The ring maintenance timer gives station management control over whether the station solicits successors immediately on entering the token ring or defers for one (or more) token pass(es). The ring_maintenance_initial value is set into the ring maintenance timer when the station enters the ring. If this value is large, the station will not find the timer expired and will solicit successors immediately. If this value is zero, the station will find the timer expired and will pass the token.

When a station gets the token, it services the four access_class data queues and then performs ring maintenance. If the inter_solicit_count is zero, the station should solicit its successors. The do_solicit_successor arc is taken if the ring maintenance token rotation timer has not expired. If either the timer has expired or the inter_solicit_count is not yet zero, the do_pass_token arc is taken and the token is simply passed.

5A.1.3.2 User notification of ring membership. It is necessary for the user of PROWAY to have access to a list of other stations which are active in the logical token ring. The creation of this live list is performed by station management as described in Part 10A.

Whenever a station changes its successor in the logical ring, the station's own station management entity will be notified. That station management entity will then read the new successor's address (NS) and inform other stations when appropriate.

5A.1.4 Token ring initialization. Initialization is primarily a special case of adding new stations; it is triggered by the exhaustion of an inactivity timer (bus_idle) in one station. If the inactivity timer expires, the station sends a claim_token frame. As in the response window algorithm, the initialization algorithm assumes that more than one station may try to initialize the network at a given instant. This is resolved by address sorting the initializers.

Each potential initializer sends a claim_token frame having an information field length that is a multiple of the system slot time (the multiple being 0, 2, 4, or 6 based on selected bits of the station address). Each initializing station then waits one slot time for its own transmission, and for other stations that chose the same frame length, to pass. The station then samples the state of the medium.

If a station senses non-silence, it knows that some other station(s) sent a longer length transmission. The station defers to those stations with the longer transmission and reenters the idle state.

If silence was detected and unused bits remain in the address string, the station attempting initialization repeats the process using the next two bits of its address to derive the length of the next transmitted frame. If all bits have been used and silence is still sensed, the station has "won” the initialization contest and now holds the token.

Once there is a unique token in the network, the logical ring builds by way of the response window process previously described.

NOTE: A random pair of bits is used at the end of the address sort algorithm to ensure that two stations with the same address (which is a fault condition) will not permanently bring down an entire system. If the two stations don't separate (random choices identical), they both attempt to form a logical ring, and at most one of them will succeed. If they do separate (random choices are different), one will get in. In the latter case, the station which doesn't get in will hear a transmission from a station with an identical address and so will discover the error condition.

5A.1.4.1 Leaving the token ring. A station may remove itself from the logical ring at any time by waiting for the token, then sending a set _ successor frame to its predecessor on the logical ring with the address of its successor. The exiting station then sends the token as usual to its successor. Readmission to the logical ring requires one of the sequences described in 5A.1.3 and 5A.1.4.

5A.1.5 Priorities. The token-passing access method provides a priority mechanism by which higher layer data frames awaiting transmission are assigned to different "service classes", ranked or ordered by their desired transmission priority. The priority mechanism allows the MAC sublayer to provide four service classes to the PLC sublayer, and higher level protocols. The priority of each frame is determined by the "priority" specified in the request command to MAC.

The token bus access method distinguishes only four levels of priority, called "access classes". Thus there are four request queues to store frames pending transmission. The access classes are named 0, 2, 4, and 6, with 6 being the highest priority and 0 the lowest.

MAC maps the two most significant bits of the priority class requested by the PLC sublayer into a two bit priority value, which is included in the frame format field. The priority value is then mapped into the MAC access class by ignoring the least significant bit in the priority field. Thus service classes 0 and 1 correspond to access class 0, service classes 2 and 3 to access class 2, service classes 4 and 5 to access class 4, and service classes 6 and 7 to access class 6.

The service class value in the request to MAC shall be carried in the FC octet. For all stations, the rule governing the transmission of highest priority frames is that a station may not transmit consecutive frames for more than some maximum time set by station management. This time, called the hi_pri _token_ hold_time, prevents any single station from monopolizing the network. If a station has more access class 6 data frames to send than it can transmit in one hi_pri_token_hold_time period, it is prohibited from sending additional frames, except for retries, after that time has expired. It then completes any required retries and must pass the token.

A station must complete all retries before passing the token to prevent the introduction of duplicate frames. For example, the remote station could correctly receive a frame but the "ACK" frame might be lost. If the local station terminated without retrying and receiving an ACK, then the remote station would have received and processed a frame that the local station thought was lost. On the next token round, the local station would most likely retry the original transmission and a duplicate frame would be created. For a more complete discussion of lost and duplicate frames see Appendix 3-A.

When a station has lower access class frames to send and has time available, it may only send these frames subject to the priority system rules described in these paragraphs.

The object of the priority system is to allocate network bandwidth to the higher priority frames and only send lower priority frames when there is sufficient unused bandwidth. The network bandwidth is allocated by timing the rotation of the token around the logical ring. Each access class is assigned a "target" token rotation time. For each access class the station measures the time it takes the token to circulate around the logical ring.

If the token returns to a station in less than the target rotation time, the station can send frames of that access class until the target rotation time has expired. If the token returns after the target rotation time has been reached, the station cannot send frames of that priority on this pass of the token.

Each station shall have three rotation timers for the three lower access classes. Each access class has a queue of frames to be transmitted. When a station receives the token it first services the highest access class queue, which uses the hi_pri_token_ hold _time to control its operation. After having sent any frames of the highest priority, the station begins to service the rotation timers and queues, working from higher to lower access classes.

Each access class acts as a virtual substation in that the token is passed, internally, from the highest access class downward, through all access classes, before being passed to the station's successor.

The access class service algorithm consists of loading the residual value from a token rotation timer into a "token hold timer" and reloading the same rotation timer with the target rotation time for that access class. (Thus frames sent by a station, for this access class, are accounted for in the access class's next token rotation time computation.) If the "token hold timer" has a remaining positive value, the station can transmit frames at this access class until either the "token hold timer" times out or this access class's queue is empty. When either event occurs the station begins to service the next lower access class.

In all cases the station completes any required retries before moving to the next lower access class. When the lowest level is serviced, the station performs any required logical ring maintenance and passes the token to its successor.

5A.1.6 MAC confirmed data transfer service. Immediate response procedures, coupled with appropriate PLC procedures, provide confirmed data exchanges. When a higher layer entity requests a confirmed data transmission, the PLC entity issues a MA_DATA.request with a request_with_response frame to the local MAC entity.

When the local MAC entity obtains the token, it sends the request_with_response frame and waits for a response frame. If a timer expires without a valid response, the local MAC entity will transmit the original frame. This is repeated until a response is received or the allowed number of retries is exhausted.

When the remote (responding) MAC entity receives the request_with_response frame, it passes the received frame to the remote PLC entity. The remote PLC entity generates an appropriate response and directs the remote MAC entity to send this response frame at once to the source of the request_with_response frame (i.e., the local station).

When the local (originating) MAC entity receives the response frame or when the retry count is exhausted, it associates the frame (if available) with the request_with_response frame now being processed and notifies the originating PLC entity of the completion of its original request.

The retry mechanism of the request_with_response procedure prevents loss of frames at any arbitrary level of confidence determined by the maximum_retry_limit. However, since the local station repeats the request_with_response frame when a response frame is not received, it is possible for the remote station to receive duplicates of the original request_with_response frame. The remote PLC entity eliminates duplicate frames using the procedures given in Part 3B. A more complete explanation of the causes for and prevention of loss and duplication is given in Appendix 3-A.

The local (originating) MAC engages in one request_with_response activity at a time. All retries and timeouts for that request are completed before the local MAC processes another request or passes the token.

5A.1.7 Randomized variables. Several of the variables used by the medium access protocol have two_bit "random" values. Some of these randomized variables are used to improve error recovery probabilities under certain conditions. The randomization of max_inter_solicit_count forces stations to operate "out of step" when opening response windows.

5A.2 Access control machine (ACM) states. The medium access logic in a station is described here as a computation machine which sequences through a number of distinct phases, called states. These states are introduced in the following clauses. The states and transitions between them are illustrated in Figure 5-3. (The dashed lines group states into functional areas.) Part 5C contains the complete state transition table which provides a formal description of the token-passing bus access machine.

5A.2.0 Offline. "Offline" is the state the access machine is in immediately following power-up or following the detection, by the MAC sublayer, of certain fault conditions. After powering up, a station tests itself and its connection to the medium without transmitting on the medium. This "internal" self testing is station implementation dependent and does not affect other stations on the network. Thus, the self-test procedure is beyond the scope of this standard.

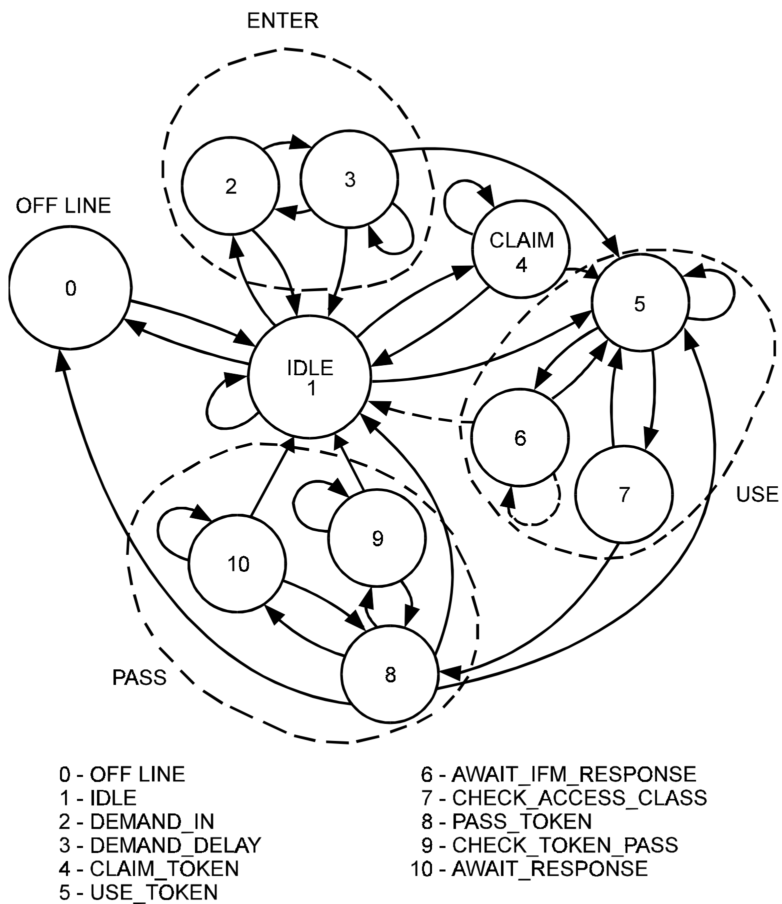


Figure 5-3: MAC finite state machine diagram

After completing any power-up procedures, the station remains in the offline state until it has had all necessary internal parameters initialized and been instructed to go online.

5A.2.1 Idle. "Idle" is the state where the station is listening to the medium and not transmitting.

If a MAC_control frame is received for which the station must take action, the appropriate state is entered. For example, if a token frame, addressed to the station, is received, the station enters the "use token" state.

If the station goes for a long period of time (a defined multiple of the slot_time) without hearing any activity on the medium, it may infer that recovery of the logical ring is necessary. The station attempts to claim the token (enters the "claiming token" state) and (re)initializes the logical ring.

5A.2.2 Demand-in. The "demand-in" state is entered from the idle state if a solicit_successor frame that spans the station's address is received by a station desiring logical ring entry. (The demand-in state is also entered from the demand-delay state during the contention resolution process discussed in 5A.1.3.) In the demand-in state the contending station sends the token holder a set_successor frame and goes to the demand-delay state to await a response.

If a station intends to respond to a solicit_successor frame or a who_follows in the first response window, the station enters the demand-in state from the idle state with a zero delay and so immediately transmits a set_successor response and goes to the demand-delay state. If the station intends to respond in the second response window after a frame or is participating in the contention resolution process, the station delays in the demand-in state before transmitting a set_successor frame.

While delaying in the demand-in state, if the station hears any transmissions, it must assume that another station with a higher numbered address is requesting the token, and so it must return to the idle state.

5A.2.3 Demand-delay. "Demand-delay" is the state the station enters after having sent a set_successor frame in the demand-in state. In the demand-delay state a station can expect to hear:

1) A token from the token holder indicating its set_successor frame was heard,

2) A resolve_contention frame from the token holder indicating that all stations which are still demanding into the logical ring should perform another step of the contention resolution process, or

3) Set_successor frames from other stations, which the station ignores.

If the station either hears nothing or hears a frame other than one of the above, the station must leave the demand-delay state. The station then abandons soliciting the token and returns to the idle state.

In case 1 above, the token holder has heard the soliciting station and sent the token. The contention resolution process is over. The soliciting station upon receiving the token goes to the use token state and begins transmitting.

In case 2, the token holder has heard responses from multiple stations soliciting the token and sent a resolve_contention frame. All stations currently in the demand-delay state respond to this frame. The responding stations first set a delay and return to the demand-in state where they listen for other contenders. If no other contenders have been heard by the time the delay period has expired, the station then sends another set_successor frame to the token holder.

The contention for the token by multiple stations is resolved by having each station delay a period of time in the demand-in state before transmitting another set_successor frame. The delay interval is chosen by taking the station's (unique) address and using two bits from that address to determine the delay interval. The first pass of the resolution process uses the most significant two address bits; the next pass the next two address bits; etc. Thus stations will delay 0, 1, 2, or 3 slot times when entering the demand-in state before transmitting.

When multiple stations request to enter the logical ring, the desired result is for the token holding station to pass the token to the highest addressed station. In order to select the highest addressed contending station from multiple contenders, the one's complement of the station's address is used to determine the delay in the demand-in state. Thus, stations with numerically higher addresses delay shorter intervals and send their set_successor messages sooner than stations with lower addresses. Stations with numerically lower addresses hear the transmissions of the higher addressed stations and drop out of the contention process.

If two contending stations have the same value for the selected two address bits, they delay the same amount of time and transmit more or less simultaneously. If the token holder hears multiple responses and does not hear a valid set_successor from any station, it sends another resolve_contention frame, starting another step of the contention resolution process.

The contention resolution process can take at most nine passes (16/2+1) for 16-bit addresses of the following cycle:

1) All remaining demanding stations send set_successor frames to the token holder.

2) They all listen for a response from the token holder and ignore other set_successor frames.

3) They all hear a resolve_contention frame from the token holder.

4) They all delay a number of slot times based on the next two bits of their own addresses.

5) If they hear another frame during the delay, they drop from contention.

The contention resolution process should resolve so that the contending station with the highest address is heard by the token holder and receives the token. However, if two stations are erroneously assigned the same station address, they will both sequence through the contention process using the same delays and may not resolve.

To permit eventual resolution in this error condition, a final resolution pass is taken, using a two-bit random number, after the station's address bits have all been used and the contention is still unresolved. If both stations choose the same random value or another error prevents resolution, then the token holder and the contending stations abandon the resolution process until the next response window is opened.

(Thus, two stations with the same address which consistently choose the same random number value may never be able to enter the logical ring.)

5A.2.4 Claim_token. The "claim_token" state is entered from the idle state after the inactivity timer expires (and the station desires to be included in the logical ring). In this state, the station attempts to initialize or reinitialize the logical ring by sending claim_token frames.

To resolve multiple simultaneous stations sending claim_token frames, each station delays for one slot_time after sending the claim_token frame, and then monitors the medium as previously described. If after this delay the bus is quiet, the station sends another claim_token frame.

If the station sends max_pass_count (where max_pass_count is half the number of bits in the station's address plus one, since the address bits are used in pairs) claim_token frames without hearing other transmissions, the station has successfully "claimed" the token, and goes to the use-token state.

5A.2.5 Use_token. Use_token is the state the station enters after just receiving or claiming a token. This is the state in which a station can send data frames.

Upon entering this state just after receiving or claiming a token, the station starts the "token holding" timer, which limits the amount of time the station may begin transmission of frames before passing to the next access class. The expiration of the token holding timer does not prevent retries, and all retries must be completed before passing to the next access class. The value initially loaded in the holding timer, hi_pri_token_hold_time, is a system_imposed parameter.

After each data frame is sent, the ACM enters the await_IFM_response state. It will return to the use_token state as soon as the confirmation requirements of the frame last sent are satisfied.

When a station's token holding timer has expired and all required retries are initiated and any transmission in progress is complete, or when the station has no more to transmit, it enters the check_access_class state.

When a station sends a data frame it sets the response_ window _timer to 3 slot times and enters the await_IFM_response state.

5A.2.6 Await_IFM_response. This state is entered when a data frame has been sent. The ACM may wait for the Interface Machine (IFM) to signal the reception of a response.

If the frame sent in the use_token state was a request_with_no_ response frame, no response is expected. The use_token state is again entered to check for another frame or holding timer timeout. If the frame sent was a request_with_response frame, the station waits in the await_IFM_response state for one of the following:

1) A response frame addressed to the requester

2) Any other valid frame

3) A timeout

If a response frame addressed to the requester is heard, the use_token state is again entered to check for another frame or holding-timer timeout. (The IFM passes the response frame to the PLC entity as it does all other data frames addressed to the station. The IFM also associates the response frame with the request_with_response frame just previously transmitted.)

If any other valid frame is heard, an error has occurred. The station returns to the idle state and processes the received frame.

If a timeout occurs before a valid frame is heard, the station repeats sending the request_with_response data frame. If the station repeats sending that frame the number of times specified by the maximum_retry_limit, a parameter established by station/management, then the request is abandoned, and the IFM notifies the PLC entity that no response to the frame was received. The use_token state is entered to check for another frame or token_holding_timer timeout.

5A.2.7 Check_access_class. "Check_access_class" controls the transmission of frames for different access classes. If the priority option is not implemented, all frames are considered to be high priority and the check_access_class state only serves to control entry to token passing.

A station may send frames for lower access classes before passing the token. Each access class other than the highest has a target token rotation time. At the time that the station has the token and begins to consider transmitting frames for that access class, the residual time left in the target rotation timer is loaded into the token holding timer and the station transitions back to the use_token state. At this time the target rotation timer is also reloaded to its initial value.

Thus the station will alternate between use_token and check_access_class states for each access class. If time is available, data frames will be sent in the use_token state. When the lowest priority access class has been checked, the station will proceed to the pass_token process, described next.

After a station has completed sending data frames, it must enter the token passing state. Three conditions can occur

1) The station knows its successor, so simply passes the token and enters the check_token_pass state.

2) The station knows its successor but must first check if new stations desire to enter the logical ring. The station sends a solicit_successor frame and enters the await_response state.

3) The station does not know its successor. (This condition occurs after initialization and under error conditions.) The station sends a solicit_successor frame, opening response windows for all stations in the system, and enters the await_response state.

5A.2.8 Pass_token. "Pass_token" is the state in which a station attempts to pass the token to its successor.

Before passing the token, the station will, when its inter_solicit_count value is zero and the time remains on its ring_maintenance timer, allow new stations to enter the logical ring. The token holding station does this by sending a solicit_successor_1 or solicit_successor_2 frame, as appropriate, and enters the await_response state. (See the description of response windows for the details of this operation.)

Following any new successor solicitation if the address of the successor, NS, is known, the station performs a simple token pass. (See the description of token passing for details of this operation.) If the successor responds and the station hears a valid frame, the station has completed its token passing obligations.

If NS is not known, the station sends a solicit_successor_2 frame to itself. Since this frame has two response windows and identical source and destination addresses, it forces all stations on the network which desire to be in the logical ring (whether or not they were previously) to respond. Those stations whose addresses are lower than the sender of the token frame transmit in the first window; those with addresses higher transmit in the second.

The station monitors the response windows for set_successor frames from potential successors, exactly as for the token pass. If no responses are heard, the station stops trying to maintain the logical ring and listens for transmissions from any other station. (See the description of token passing for details of this operation.)

5A.2.9 Check_token_pass. "Check_token_pass" is the state in which the station waits for a reaction from the station to which it just passed the token.

The station sending the token waits one slot time for the station receiving the token to transmit. The one slot time delay accounts for the time delay between receiving a frame and the recipient taking the response action.

If a valid frame is heard which started during the response window, the station assumes the token pass is successful. The frame is processed as if it were received in the idle state.

If nothing is heard in one slot time, the station sending the token assumes the token pass was unsuccessful and returns to the pass_token state to either repeat the pass or try another strategy.

If noise or an invalid frame is heard, the station will continue to listen for additional transmissions, as described in detail in 5A.1.2.

5A.2.10 Await_response. In the await_response state the station attempts to sequence candidate successors through a distributed contention resolution algorithm until one of those successors' set_successor frames is correctly received or until no successors appear. The state is entered from the pass_token state whenever the station determines it is time to open a response window or if the station does not know its successor (as in initialization or when a token pass fails).

The station waits in the await_response state for a number of response window times. If nothing is heard for the entire duration of the window(s) opened, the station goes to the pass_token state, either to pass the token to its known successor or to try a different token-pass strategy.

If a set_successor frame is received, the station waits for the rest of the response window time to pass. The station then enters the pass_token state and sends the token to the new successor.

If the received frame is other than a set_successor frame, the station drops the token (since someone else must have it to be able to send any other frame type, thus a duplicate token situation) and reenters the idle state.

If noise is heard during the response windows, the station cycles through a procedure of sending resolve_contention frames which open four response windows each, and waiting for a distinguishable response that began in a response window. The loop repeats a maximum of max_pass_count times, each time instructing contending stations to select a different two bits of their address to determine which of the four opened windows to transmit in.

5B. MAC sublayer definitions and requirements

Part 5B specifies mandatory aspects of the MAC sublayer operation and mechanism which are not specified in Part 5C or 5D. All specifications in Part 5B are required for conformance to this standard.

5B.1 MAC definitions. The following paragraphs define critical MAC parameters, which are constrained by this specification.

5B.1.1 Immediate_response. Immediate_response is defined as the immediate transmission of a response to a received frame. This assumes that no other transmissions or actions were intervening.

5B.1.2 MAC-symbols. MAC-symbols are defined as the smallest unit of information exchanged between MAC sublayer entities. The six MAC-symbols are as follows:

Name	Abbreviation
zero	0
one	1
non_data	N
pad_idle	P
silence	S
bad_signal	B

Where binary 0 and 1 data bits are discussed, they are sent and received as **zero** and **one** MAC-symbols, respectively.

5B.1.3 MAC_symbol_time. MAC_symbol_time is the time required to send a single MAC-symbol. This is the inverse of the LAN data rate.

Nominal Data Rate	Nominal MAC_symbol_time
1 M bits/sec	1 microsec
5 M bits/sec	200 nanosec
10 M bits/sec	100 nanosec

5B.1.4 Octet_time. The term octet_time shall be understood to correspond to the time interval required to transmit eight (8) MAC-symbols.

5B.1.5 PHY-symbols. Physical layer symbols (PHY-symbols) correspond to the waveforms impressed on the physical medium. (See Section 8.2.1.1 for the PHY-symbol definitions).

5B.1.6 Transmission_path_delay. Transmission_path_delay is the worst case delay which transmissions experience going through the physical medium from a transmitter to a receiver. The following formula is a general definition of transmission_path_delay:

Transmission_path_delay = worst_case physical_medium_delay

NOTE: Refer to Section 9.7 for a detailed discussion of transmission_path_delay.

5B.1.7 MAC_station_delay. MAC_station_delay is the time from the receipt of the PHY-symbols corresponding to the last MAC-symbol of the received ED at the receiving station's physical medium interface to the impression of the first immediate_response PHY-symbols onto the physical medium by that station's transmitter when the frame is a MAC frame.

5B.1.8 PLC_station_delay. PLC_station_delay is the time from the receipt of the PHY-symbols corresponding to the last MAC-symbol of the received ED at the receiving station's physical medium interface to the impression of the first immediate_response PHY-symbols onto the physical medium by that station's transmitter when the frame is a request_with_response data frame.

PLC_station_delay shall be ≤ 16 microseconds.

5B.1.9 Safety_margin. Safety_margin is defined as a time interval no less than one MAC_symbol_time.

Safety_margin ≥ MAC_symbol_time

5B.1.10 Slot_time. Slot_time is the maximum time any station need wait for an immediate_response from another station. Slot_time is measured in octet_times, and is defined as

Slot_time = INTEGER ({[2*(Transmission_path_delay + MAC_station_ delay)
+ Safety_margin] / MAC_symbol_time
+ 7} / 8)

5B.1.11 Response_time. Response_time is the anticipated time any station need wait for a response_frame from another station. Response_time is defined as

Response_time = slot_time + PLC_station_delay + nominal preamble

The response_time determines the maximum amount of time the ACM may spend in state six (await_IFM_response) as shown in Figure 5-3.

5B.1.12 Response_window. A response_window is the basic time interval which the MAC protocol allows, following certain MAC_control frames, for an immediate_response from another station. This interval is one slot_time long.

Response_window duration = slot_time

If a station, waiting for a response, hears a transmission start during a response_window, that station shall not transmit again at least until the received transmission terminates.

5B.1.13 Maximum_retry_limit. This limit is the maximum number of retries a MAC sublayer will make in attempting to secure a response to a request_with_response frame.

5B.2 Transmission order. The frame formats used by the MAC sublayer and the detailed contents of the octets of those frames are specified in Part 5D. The octets of a frame and the MAC-symbols of an octet shall be transmitted from the PLC to the MAC sublayer, and from the MAC sublayer to the Physical layer, and vice versa, in the order specified in Figure 5-7; the first octet of the frame shall be transmitted first and the first MAC-symbol of each octet shall be transmitted first. The first octet and first MAC-symbol correspond to the top octet and the left-most MAC-symbol shown in Figure 5-7. Those notations within octets of Figure 5-7 which are not MAC-symbols are casual descriptions.

Figure 5-7 describes a complete general frame or MAC protocol_data_unit (M_pdu) containing a PLC protocol_data_unit (L_pdu).

5B.3 Delay labeling. Vendors shall provide a worst case value for MAC and PLC station delay times. Vendors must also specify a minimum network slot time when their equipment anticipates some minimum delay in order to function correctly. When uncertain of the exact value of the delay, vendors shall state an upper bound. Vendors of equipment conforming to this standard shall label the equipment with that equipment's contribution to the station delay. A vendor of a complete station would label the station with the total station delay. A vendor of a component, intended to be assembled by an end user into a station, would label the component or otherwise document the delays that contribute to station delay.

5B.4 Miscellaneous requirements

5B.4.1 Station initialization. On power-up, the station shall enter the offline state. While in the offline state the station shall not impress any signaling on the LAN medium.

The station shall progress from the offline state to the idle state only when it has been loaded with the basic station operating parameters needed for correct operation of the MAC protocols. These operating parameters include at least the following:

1) TS (station address)

2) address_length (implicit in TS)

3) slot_time

4) hi_pri_token_hold_time

5) max_ac_4_rotation_time

6) max_ac_2_rotation_time

7) max_ac_0_rotation_time

8) max_inter_solicit_count

9) max_ring_maintenance_rotation_time

10) ring_maintenance_timer_initial_value

11) maximum_retry_limit

12) in_ring_desired

13) min_post_silence_preamble_length

5B.4.2 Token-passing order. The token shall be passed from station to station in numerically descending station address order, except that the station with the lowest address shall pass the token to the station with the highest address, in order to close the logical ring. Figure 5-8 illustrates the address-ordered logical ring and shows the logical relationships which hold between addresses of adjacent stations in a logical ring with three or more members.

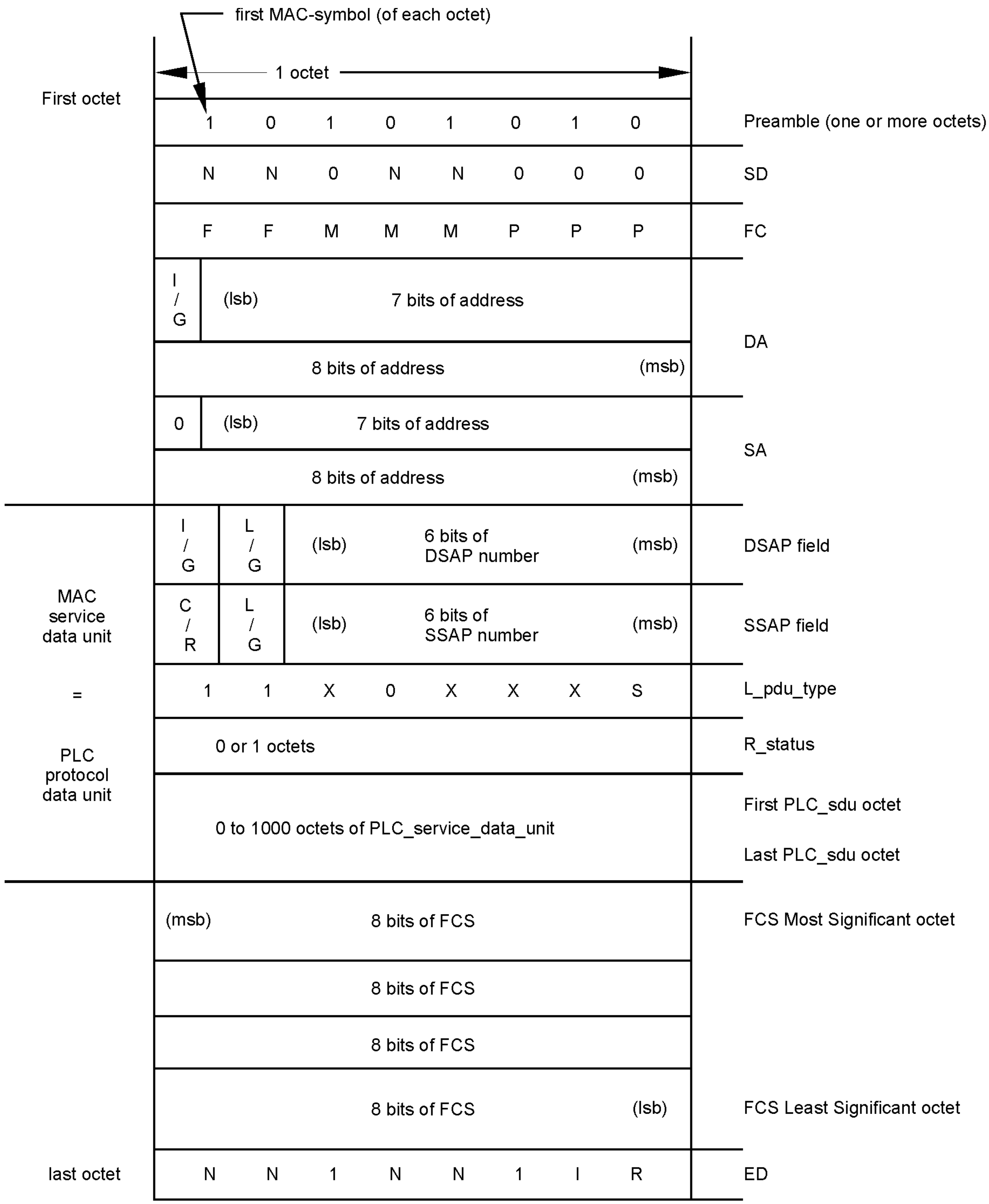


Figure 5-7: PROWAY PLC and MAC Protocol_Data_Unit composition and transmission order with 16-bit address

5B.4.3 Station receipt of its own transmission. In systems with significant transmission_path_delay, a transmitting station may receive its own transmissions after some small but significant delay. The MAC access mechanism of such a transmitting station shall not be misled by the receipt of its own transmissions. The state diagrams in Part 5C specify where a station should ignore its own transmissions.

5B.4.4 Token holding time. The token holding station shall only begin transmitting a non-retry frame when there is time remaining on the token_hold_timer. A transmission may run past the expiration of the token_hold_timer by up to the time necessary to transmit a maximum length data frame. Waiting for any response frame and any required retries may also cause a token holding station to exceed the token_hold_timer.

5B.4.5 Address lengths. Addresses shall be two octets (16 bits) or optionally 6 octets (48 bits) long. Address composition is specified in Part 5D, and examples are given in Appendix 5-B. Note that use of 48-bit addresses may reduce system throughput and responsiveness.

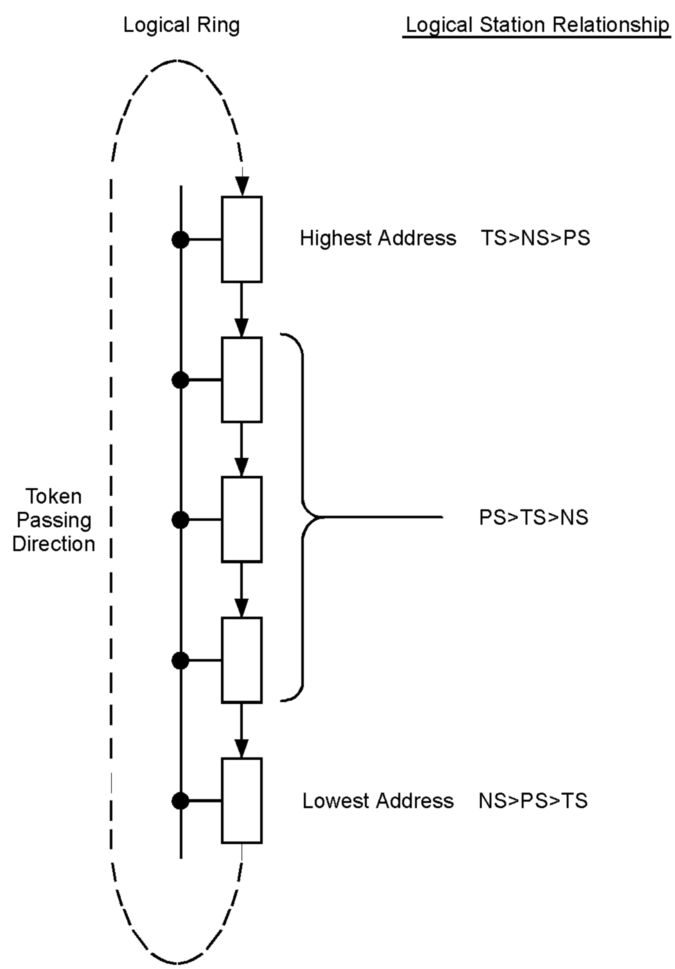


Figure 5-8: Logical token-passing ring

5B.4.6 Randomized variables. The station must provide a two-bit (that is, four valued) random variable for use in the MAC protocols. For the medium access protocol to benefit from the randomization, the technique used to create the random values must be statistically independent between stations. Thus random number generators tied in any way to the network data clock, for example, would not produce statistically independent variable values.

The variables shall be re-randomized "periodically". Periodically shall be interpreted to mean either an interval not to exceed 50 milliseconds, or every use of the random variable.

5B.4.7 Contention delay. If the station hears a solicit_successor or who_follows frame, it determines which response window in which to contend based on the station's address and the SA/DA addresses in the frame. If the station wants to contend in the first window, it loads the contention timer with zero, so the station proceeds immediately to the demand-in state. If the station wants to contend in the second window, the contention timer is loaded with one, so the station listens during the first window.

Following receiving a resolve_contention frame, if the station is contending, it loads the contention timer with the one's complement of two bits selected from its own address as indexed by the resolution pass count. The station thus listens zero, one, two, or three slot times before again contending.

5B.4.8 Token claiming. If the bus_idle_timer expires, a station may transmit a claim_token frame and set the claim_timer. When the claim_timer expires, if no transmissions are present at that instant, the station sends an additional claim_token frame and repeats the delay and transmission check. This procedure is repeated until either transmissions from another station are heard or the value of the claim_pass_count equals max_claim_pass_count.

The length of the claim_token frames are 0+, 2+, 4+, or 6+ slot_times as a function of two bits of the station's address. Indexing through the address performs an address sort in the claim process, leaving the station with the highest address claiming the token.

5B.5 Use of address bits in contention algorithms. The contention processes used to claim a new token or demand logical ring entry both use the bits of the station address to accomplish a sorting-like resolution in which the station having the numerically largest or highest address value wins. The following paragraphs treat the address as an array of binary (0/1) values or address bits; for notational purposes, "address(*i*)" indicates the *i* th binary bit of the station's address, with address(1) the most significant bit. These address_bits are used two at a time, starting with the most significant address_bits.

5B.5.1 Claim_token frame length. A station which is attempting to claim a new token first determines that no other station is transmitting, then transmits a claim_token frame containing a data_unit with a length equal to 0, 2, 4, or 6 slot_times. It then waits or delays one slot_time before again listening for other transmissions. The token claiming contention process shall consist of *N* cycles of listening, transmitting and delaying, where *N* is a function of the station's address length in bits:

N = (address_length / 2) + 1

The length, L, of the n th claim_token frame's data_unit, in octet_times (for the n th cycle of the token claiming process), shall be determined as follows:

for $1 \leq n < N$

L: = 2 * slot_time * bit_value

bit_value: = (2*address_bit ((2 * n) -1) + address_bit (2 * n)

where bit_value equals 0, 1, 2, or 3 as a function of the two bits used in cycle n.

for $n = N$

$: = 2 * \text{slot\_time} * \text{random\_4}$

where random_4 = a random number equal to 0, 1, 2 or 3.

5B.5.2 Demand delay time interval. A station which is demanding entry into the logical ring first listens for other transmissions, delaying its next transmission for 0, 1, 2 or 3 slot_times; then, if it has heard no other transmissions, it transmits a fixed length set_successor frame. This delay preceding set_successor frame transmissions is called the demand-delay. The contention process for demanding logical ring entry shall consist of at most *N* cycles of transmission and listening delays, where *N* is a function of the address length in bits:

$N = (\text{address\_length} / 2) + 1$

The number of slot_times, *D*, to delay after the *n* th transmission (for the *n* th cycle of the contention resolution process) shall be determined as follows:

for $1 \leq n < N$

$D: = 2 * \text{slot\_time} * \text{bit\_value}$

$\text{bit\_value}: = (2 * \text{address\_bit}\ ((2 * n)-1) + \text{address\_bit}\ (2 * n))$

where bit_value equals 0, 1, 2 or 3 as a function of the two bits used in cycle *n*.

for $n = N$

$D: = \text{random\_4}$

where random_4 = a random number equal to 0, 1, 2 or 3.

5B.6 Priority of transmitted frames

5B.6.1 Access classes. The priority mechanism shall provide four levels of service with respect to a frame's priority of access to the medium; these levels are called access_classes. The access_classes shall be identified as 0, 2, 4 and 6, and access_class 6 shall be the highest priority or most favored level of service.

5B.6.2 Priority to access_class mapping. The PLC request priority contained in the request to MAC is eventually satisfied through use of the access_classes. The priority request shall first be satisfied by assignment of the request to one of eight MAC service_classes. These MAC service_classes shall then be mapped into MAC access_classes as described in the following table:

service_class	access_class	priority
0, 1	0	lowest
2, 3	2	
4, 5	4	
6, 7	6	highest

5B.6.3 Token_rotation_timers. A station shall provide three (actual or virtual) token_rotation_timers, one for each access_class. These timers shall all run concurrently, counting downward from an initial value to zero, at which point they shall stop counting and their status shall be "expired".

These timers shall count in units of octet_times and shall otherwise be managed as specified in Part 10C.

5B.6.4 Recommended priority assignments in PROWAY system. The following priority assignments are recommended for all PROWAY systems:

access_class	usage
6	Urgent messages, i.e., those performing critical alarm, interlock and coordination functions.
4	Normal control actions and ring maintenance functions.
2	Routine data gathering and display and data base updates.
0	File and program transfer.

5B.7 Delegation of right to transmit. A station holding a token may request a second station to transmit a response without the second station holding the token. The first station, in effect, delegates the authority to transmit to a secondary station.

The secondary station shall conform to all of the requirements imposed by this standard on the token holder except for participating in the logical token passing ring and associated protocol mechanisms (unless the secondary is or needs to be in_ring). The secondary station need not be in the logical ring from an address sense. The secondary shall not transmit on the network unless

1) Delegated as a transmitter by a token holder, or

2) Transmitting is authorized by the procedures specified in Part 5C.

5B.8 Notification of logical ring modification. A change in the Next_Station (NS) variable requires that the MAC sublayer generates an indication to station management. Station management typically uses this to participate in the maintenance of a list of active stations.

5B.9 Station addresses. Station addresses shall be defined as given in 5D.1.4.1.2 and 5D.1.4.2.

5B.10 Required MAC machines. All MAC machines are required for **INITIATOR** stations. All MAC machines except the ACM are required for **RESPONDER** stations.

5B.10.1 Mandatory access control machine (ACM) functions. All ACM functions are mandatory in all implementations of **INITIATOR** stations.

5B.11 Interface machine

5B.11.1 Interface machine (IFM) description. The Interface Machine (IFM) acts as an intermediary between the other functional machines of the MAC sublayer and the PLC and LLC sublayers and station management entity with which MAC must communicate.

The IFM has eight primary functions:

1) Accepting or generating the service primitives supported at the MAC-MAC_user interface.

2) Mapping higher layer requests between MAC_user terms of quality of service (priority and confirmation class) and MAC terms (access_class and MAC_action).

3) Queuing pending service requests into one of four queues, separated by access_class, so that the requests may be handled according to access_class, as well as according to arrival order. Requests from PLC, LLC and station management at a given access_class share the same queue and are serviced on a first-in first-out basis.

4) Recognizing the individual destination address of data frames destined for this station and the broadcast address.

5) Recognizing group addresses.

6) Notifying the ACM that a response frame has been received (for request_with_response frames) and passing this frame to the local (originating) PLC entity in association to the corresponding confirmation.

7) Passing received request with_no_response and request_with_response frames to the destination MAC_user entity specified by frame_type.

8) Accepting a response frame from the remote (responding) PLC entity following receipt of a request_with_response frame, and sending this response frame.

5B.11.2 Mandatory interface machine (IFM) functions. IFM functions that are mandatory for **INITIATOR** and **RESPONDER** stations are indicated by an M in the appropriate column in the table below. Optional functions are indicated by an O. A hyphen (—) in this table indicates that the associated function is not required for the corresponding type of station. These functions are fully described in Section 5B.11.1.

FUNCTION	INITIATOR	RESPONDER
1) Accepting or generating supported service primitives	M	M
2) Mapping between MAC_user and MAC terms	M	M
3) Queuing service requires into queues separated by access_class	M	_
4) Recognizing this station's individual destination address and the broadcast address	M	M
5) Recognizing group addresses	M	O
6) Receiving, associating, and passing response frames to the originating MAC_user	M	_
7) Passing request_with_response and request_with_no_response frames to the specified MAC_user	M	M
8) Accepting and transmitting response frames	M	M

The minimal attributes of MAC entities for **INITIATOR** and **RESPONDER** stations are

Priorities supported	4	4
Group addresses recognized	16	0

5B.12 Receive machine

5B.12.1 Receive machine description. The receive machine accepts MAC-symbols (see 5B.1.2) from the PHY layer and generates high-level data structures and signals for the MAC access control machine and the MAC interface machine (IFM).

The receive machine is specified in Reference 4, Appendix I-A.

The interface between the PHY layer and the MAC access machine is the PHY_DATA.indication primitive specified in Part 6. The description of the receive machine in Reference 4 embodies the PHY_DATA.indication primitive as an encoded MAC-symbol and an associated PHY_clk.

5B.12.2 Mandatory receive machine (RxM) functions. All RxM functions are mandatory in all implementations.

5B.13 Transmit machine

5B.13.1 Transmit machine description. The media_access_control machine forwards frames for transmission to the TxM as a unit (at least for the purposes of this description). The TxM then passes the data frame, along with appropriate delimitation, to the physical layer, one MAC-symbol at a time, for transmission on the physical medium. The transmit machine is responsible for sending the proper amount of preamble, computing the FCS and including it within the transmitted frame, and delimiting the frame with SD and ED.

The transmit machine is specified in Reference 4, Appendix I-A.

5B.13.2 Mandatory transmit machine (TxM) functions. All TxM functions are mandatory in all implementations.

5C. Access Control Machine (ACM)

Formal description. This part defines the token bus medium access control mechanism, and also defines the ACM, variables and functions used in the definition of the ACM. A formal state machine description of the access control mechanism using the variables and functions discussed here is given in Reference 4, Appendix I-A.

5C.1 Variables and functions. The variables and functions of the state machine description given in Reference 4, Appendix I-A are grouped into categories as follows:

1) Variables defined by station management
2) Variables defined by the interface machine
3) Timers
4) Variables defined by the receive machine
5) Other ACM variables and functions

5C.1.1 Station management variables. Station management provides the MAC machine with the station's address bit string (and thus implicitly with the length of all addresses). Also supplied by station management are other network parameters:

TS: This station's address. A bit-string variable set to the value of the station's 16-bit or 48-bit address.

slot_time: An integer in the range of 1 to (2^{13}-1) octet_times. See 5A.1.1 and 5B.1.10.

max_pass_count: An integer equal to half the station's address length in bits, plus one. (Thus equal to 9 for a 16-bit address length.)

The value of max_pass_count limits loops in the ACM. The value is used to limit the token contention process. After cycling through max_pass_count contention cycles the process must be stopped if, due to an error, a single contender cannot be resolved.

The value of max_pass_count is also used to stop the token claiming process. After sending max_pass_count claim token frames, if no other station is heard, a station can claim the token.

min_post_silence_preamble_length: An integer equal to the minimum number of octets of preamble to be transmitted at the beginning of a transmission after the station has been silent. The value of min_post_silence_preamble_length is determined by the type of Physical layer used in the station.

max_inter_solicit_count: An integer number of token possessions within the range 16 to 255. The value, in addition to the ring_maintenance_timer, determines how often a station opens response windows. Normally, a station opens response windows prior to every *n* th pass of the token, where *N* is the value of max_inter_solicit count. The action is taken only when token rotation time has not exceeded the ring_maintenance_target_time.

If all stations in the ring used the same max_inter_solicit_ count, they would all consistently open response windows on the same token rotation. This action could lead to rapid token rotations where no response windows were opened, and occasional rotations where every station opened a response window before passing the token.

To avoid all stations in the ring having the same value of max_inter_solicit_count, the least significant two bits of the value shall be chosen randomly. The actual value used for the max_inter_solicit_count shall be changed by each station by re-randomizing the least significant two bits of the variable at least every 50 milliseconds or on every use.

target_rotation_time (access_class): An array of integers in the range 0 to 2^{21} -1 octet_times, used with the priority option and with the ring_maintenance timer. (See 5C.1.4 for a discussion of the variable's function.)

ring_maintenance_timer_initial_value: An integer in the range 0 to 2^{21} -1 octet_times, used to determine the initial value of the ring_maintenance token_rotation_timer upon entry to the ring. A large value will cause the station to solicit successors immediately upon entry to the ring; a value of zero will cause the station to defer this solicitation for at least one rotation of the token.

hi_pri_token_hold_time: An integer in the range of 0 to 2^{16} -1 octet_times. Used to control the maximum time a station can transmit frames at access class 6.

in_ring_desired: A Boolean variable which determines the access control machine's steady-state condition when it has no queued transmission requests. If the variable is true, the station should be in_ring (a participant in the token-passing logical ring). If false, the station should be out_of_ring (an observer of the token-passing logical ring).

maximum_retry_limit: An integer constant in the range of 0 to 7 specifying the maximum number of attempts that will be made to send a request_with_response frame.

5C.1.2 Interface machine variables and functions

get_pending_frame (access_class): A function provided by the interface machine of MAC. This function pops the first frame off the pending frame queue for the indicated access class, and returns it to the access control machine for transmission.

any_send_pending: A Boolean variable reflecting the logical OR of all the Boolean variables send_pending (access_class). Any_send_pending is true if at least one of the pending frame queues is non-empty. If all queues are empty, the variable's value is false.

power_OK: A Boolean variable indicating that the ACM may begin operation. Provided by station management hardware.

5C.1.3 Note on logical ring membership control. The two Boolean variables in_ring_desired and any_send_pending determine the operation of the ACM with respect to contending for the token and being in the logical ring as follows:

Variables and States		ACM Actions
in_ring_desired	any_send_pending	
false	false	Do not contend for token. Drop out if currently in token-passing logical ring.
false	true	Contend for token. Send data, which may empty the ending frame queues and take any_send_pending false. Exit logical ring if any_send_pending becomes false.
true	false	Contend for token if not sole active station. Remain in token-passing logical ring even without data to send.
true	true	Contend for token. Remain in token-passing logical ring and send data.

5C.1.4 Timers. A number of timers are used in the description of the state machine. A timer is expressed as a set of procedures and a Boolean variable. The procedures are named xx_timer.start (value), where xx is the timer name and value is an integer that sets the timer delay. The xx_timer.value returns the current value of the counter. The Boolean variables are named xx_timer.expired and have a value of false while the timer is running and true when the timer has expired.

For example, the bus_idle timer would be set to a value of one (slot_time) by executing bus_idle_timer.start (1). The variable bus_idle_timer.expired would then be false for one slot time.

5C.1.4.1 Slot_time interval timers. The first five timers (bus_idle_timer, contention_timer, claim_timer, response_timer and token_pass_timer) work in integral multiples of the network slot_time. (The first five timers are not used concurrently, thus they could be implemented in a single hardware timer.)

bus_idle_timer. Controls how long a station listens in the idle state for any data on the medium before entering the claiming token state and reinitializing the network. Most stations wait seven slot times. The one station in the network having lowest_address true waits six slot times. The function max_bus_idle returns the value 6 or 7 depending on the state of lowest_address.

claim_timer. Controls how long a station listens between sending claim_token frames. The claim_timer is always loaded with the value 1.

response_window_timer: Controls how long a station which has opened response windows listens before transmitting its next frame.

When sending a solicit_successor, who_follows or resolve_contention frame, this timer controls the length of time a station solicits responses. After sending a solicit_successor frame, the sending station loads the response_window timer with the number of windows opened. The timer thus determines how long the station remains in the await_response state listening for stations to respond. If the timer expires and nothing is heard, the station goes to the pass_token state and passes the token to its successor.

When sending a request_with_response data frame, the response_window timer controls how long a station waits for a response frame before repeating the request_with_response frame.

contention_timer. Controls how long a station listens in the demand-in state after hearing a resolve_contention, solicit_successor or who_follows frame when the station wants to contend for the token. If the station hears a transmission while listening, it has lost the contention and must return to the idle state.

token_pass_timer. Controls how long a station listens after passing the token to its successor.

If any frame is heard before the token_pass_timer expires, the station assumes that its successor has accepted the token. If the timer expires and a frame is not heard, the station assumes its successor did not accept the token and sequences to the next stage of the pass token procedure.

5C.1.4.2 Octet_time interval timers. The remaining timers have a granularity of one octet transmission time, rather than one network slot_time. They are used to implement the access class structure and limit the time during which a station may start to transmit frames for each access class.

token_rotation_timer (access_class): This is a set of four timers, one for each of the lower three access classes and one for ring maintenance.

When the station begins processing the token at a given access class the associated timer is reloaded with the value of the target_rotation_time for that level. When the station again receives the token it may send data of that access class until the residual time in the associated token_rotation_timer has expired.

Upon initial entry to the ring, the first three priority timers are set to a value of zero (expired), and the ring_maintenance timer is set to the value ring_maintenance_timer_initial_value.

token_hold_timer. The residual time from the current token_rotation_timer is loaded into the token_hold_timer just before the token_rotation_timer is reloaded. The station may send data frames of the corresponding access class as long as the token_hold_timer has not expired.

When the station is sending highest access class messages, the value of hi_pri_token_hold_time is loaded into the token_hold_ timer. Thus highest access class messages are limited to only a fixed number of bytes regardless of current network loading.

5C.1.5 Receive machine variables and functions. The outputs of the receive machine are several state variables and a data frame, as described in the following paragraphs.

bus_quiet: A Boolean variable which is true whenever the physical layer is reporting that silence is being received. False when something other than silence is being received. bus_quiet is set and reset by the receiver machine and is only read by the ACM.

Rx_frame: A record written by the receiver machine. The record is updated to reflect the contents of the most recently received valid frame. The major fields in the record are:

FC: The one-octet frame control field
DA: The two-octet destination address field
SA: The two-octet source address field
data_unit: The multi-octet data unit field
FCS: The four-octet frame check sequence field

Rx_protocol_frame: This signal indicates that a valid frame has been received, and that the frame type is one of the MAC protocol frame types. This signal is set by the receive machine, and it is read and cleared by the access machine.

Rx_data_frame: This signal indicates that a valid frame has been received, and that the frame type is a LLC frame or station_management frame. This signal is set by the receive machine and read by both the access control machine and the interface machine; it is cleared only by the interface machine.

noise_burst: A Boolean variable set by the receiver machine when bus_quiet goes true (the bus goes from non-silence to silence) and neither Rx_protocol_frame nor Rx_data_frame were set during the transmission (that is, no valid frame was heard). It is reset by the access control machine when the noise burst condition has been processed.

5C.1.6 Other variables and functions. The following are internal to the MAC access control machine (ACM).

TH—Token Holder's address: The address of the current token holder. A temporary buffer loaded from the SA field of a solicit_successor, who_follows or resolve_contention frame. If a set successor frame is sent by the station as part of the contention process, the DA address is taken from TH.

NS—Next Station's address: The address of a station's successor in the logical ring. NS is set when a station that does not know its successor hears a solicit_successor frame and contends for the token. The station sets NS to the value of the destination address field of the frame. (If the station successfully contends in a response window, it will receive the token and eventually pass it to the station whose address was loaded into NS.)

For example: Suppose a station with address 25 is not in the logical ring, and wants to enter. If this station hears a solicit_successor frame sent by station 30 with a DA address of 20, it will set NS to 20, the DA address in the frame. If the station contends in the window and is heard by station 30, it will be passed a token. When the station has completed sending data frames, it passes the token to its successor station 20.

The NS variable is also loaded whenever the station receives a set_successor frame addressed to it.

NOTE: Wherever the value of NS is changed, a MA_EVENT. indication is given to station management.

NOTE: Once a station thinks it knows the value of NS it no longer reloads NS when a contention window is opened spanning the station's address. The reason is that under error recovery conditions, stations will send solicit_successor_2 frames addressed to themselves that open response windows for all stations. If all stations reset their NS variables at this point, any logical ring that existed would collapse.

NS_known: A Boolean variable that indicates whether the station thinks it knows the address of its successor. NS_known is set true whenever the station receives a set_successor frame addressed to it. Normally the set_successor frame follows a successful contention, as in the previous example.

NS_known is set false whenever the station leaves the logical ring.

PS —Previous Station's address: The variable is set to the value of the source address of the last token addressed to the station.

If a who_follows frame is heard, the contents of the data field of the frame are compared with the contents of PS. If they are equal, the station responds to the who_follows request with a set_successor frame.

An example will clarify the use of PS. If a logical ring contains stations with addresses 30, 25, and 20, the station with address 20 will have 25 in its PS register, since this is the address of the station that sends it the token. If station 25 fails, when station 30 tries to send the token to station 25 it will get no response. After two tries at passing the token, station 30 sends a "who follows 25?" frame. Station 20 responds by sending a "set your successor to 20" frame. In this manner the failed station, 25, is quickly patched out of the ring.

max_access_class: An integer constant used to initialize the sequencing of the processing of the pending frame queues. The value of max_access_class is 6, the highest priority access class.

access_class: An integer which is used to sequence through the access classes while transmitting data frames.

The first, or highest priority, access_class equals the value of max_access_class (that is, 6). The variable access_class is then decremented (by 2) through all classes until less than zero, and then the station performs its ring maintenance functions and passes the token.

in_ring: A Boolean variable set true when the station receives a token frame addressed to it or when the station successfully completes the claiming token process. Set false if the station sets itself out of the ring.

sole_active_station: A Boolean variable used to mute stations having defective receivers. If a station's receiver becomes inoperative in an undetected manner, the station otherwise would disrupt the operation of the system by continually claiming the token and then soliciting a successor station.

If the sole_active_station variable is true, a station is prevented from entering the claiming token process until it has data to send. Thus a station with an inoperative receiver and no data to send will remain passively out of the ring.

If a station is a member of the ring and its receiver fails, it will be unable to hear its successor claiming the token. The station will cycle through the token passing recovery algorithm, quickly reaching the point where it has sent a solicit_successor_2 frame addressed to itself and received no response. At this point, the station sets sole_active_station true and becomes passive.

Sole_active_station is set false whenever the station hears a valid frame from another station.

lowest_address: A Boolean variable set true if the station's successor address is greater than the station's address.

At any one time there should be only one station in the logical ring with lowest_address set true. This is the station with the lowest_address of all those currently in the logical ring. When this station opens response windows during a token pass, it must open two windows. The first window is used by stations having an even lower address that wish to enter the ring. The second window is used by stations having a higher address than the recipient of the token, the station currently with the highest address in the ring.

Lowest_address is computed and set by a station whenever NS is changed.

Lowest_address is used for a second purpose unrelated to token passing. If the token-holding station fails, another station must recover the token. The bus_idle_timer is a "watchdog" timer. If no frames are heard by a station for an interval greater than this timer, the claim token process is started.

In an effort to minimize interference during the claiming process, one station is selected to use a shorter bus_idle_timer value than the other stations. This station recovers all lost token failures, except one it causes. The station with lowest_address true is always unique, so it is assigned this role.

just_had_token: A Boolean variable set true when the station passes the token and set false if the station hears a valid frame from another station.

Just_had_token is used to detect duplicate addressing failures on the network. If a station hears a valid frame with a source address equal to its own address and just_had_token is false, the station cannot have sent the frame. If such a frame is heard, the MAC sublayer notifies station management of the detection of another station on the network using the same MAC address; the MAC sublayer then enters the offline state.

heard: A three-state variable used in the await_response state. The states are:

nothing: The station has heard nothing (except its own transmissions) since beginning the resolve process.

collision: A noise burst has been heard.

successor. A valid set_successor frame has been received. At the end of the resolution period, the station will send the token to the station whose address was in the protocol data unit field of (one of) the valid set_successor frame(s).

claim_pass_count: An integer with a range from 0 to max_pass_ count. Used as an index to TS to select two bits from the station's address. The value of the selected bits (times twice the slot time) determines the length of the information field of the claim_token frame to be sent. After each claim_token frame the value of the variable claim_pass_count is incremented by one.

contend_pass_count: An integer with a range from 0 to max_pass_count. Used as an index to TS to select two bits from the station's address. The one's complement of the selected bits (times the slot time) determines the length of time a station delays in the demand-in state after receiving a resolve_contention frame. If no other frames are heard before the contention_timer expires, the station sends a set_successor frame to the token holder, increments the value of contend_pass_count, goes to the demand-delay state, and waits for the token or for another resolve_contention frame.

contention_delay (cycle): An integer function which returns the value 0, 1, 2 or 3. This value is based upon the one's complement of a pair of address bits which are indicated by the cycle in the address sort. The value is used to control the number of slot_ times the station delays before transmitting when demanding entry to the logical ring.

resolution_pass_count: An integer with a range from 0 to max_ pass_count. Used to count the number of resolve contention passes the token-holding station makes. If the counter reaches the value of max_pass_count, the resolution process is abandoned and the token passed to the station's successor.

inter_solicit_count: An integer in the range of 0 to 255. Determines when a station must open a response window. Before passing the token, the value of inter_solicit_count is checked. If the value is zero, a new successor is solicited by opening a response window. If the value is not zero, the counter is decremented and the token is passed. Whenever anything is heard during the response windows following the solicit _successor frame, the counter value is set to zero so that it will again be zero when the station next receives and passes the token. Thus receipt of a set_successor frame during a response window causes the station to reopen the response window before that next token pass.

remaining_retries: An integer in the range of 0 to maximum_ retry_limit. Used to control the number of retransmissions upon timeout of a request_with_response frame.

suppress_FCS: A Boolean variable used within the ACM to indicate that the current frame should be transmitted without having the transmitter state machine append a FCS.

transmitter_fault_count: An integer in the range of 0 to 7. Used to infer that the station's transmitter has probably failed and thus that the station's transmissions cannot be heard correctly by other stations on the network.

The value of transmitter_fault_count is incremented each time the station sequences to the end of the token contention process or fails to pass the token to any successor. Neither of these failures occurs during normal operation. The value of transmitter_fault_count is reset to zero if the station either wins the demand-in token contention process or successfully passes the token, since such an event indicates that another station correctly heard a transmission from the station.

If the value of transmitter_fault_count is incremented to max_ transmitter_fault_count, the station reports a faulty_ transmitter to station management and enters the offline state. The value of max_transmitter_fault_count is set to a maximum of 7, allowing for an occasional protocol sequencing impasse due to noise. If the station cannot enter the ring or pass the token (if already in the logical ring) seven times in a row, the inference made is that something has failed in the station, probably in the transmitter, and so the station removes itself from the logical ring.

first_time: A Boolean variable that controls processing of noise bursts in the await_response state. Set true upon entry to the state. Set false when the first noise burst is heard. If a noise burst is heard when first_time is false, the station returns to the idle state.

pass_state: A multistate variable used to control the operation of the pass_token substates. The action taken in the state depends on the value of the variable pass_state as follows (the actions are listed in the order taken by a station soliciting successors and failing:

pass_state value	action
solicit_successor	Send solicit_successor frame. Enter check_await_response_state.
pass_token	Send token to successor. Enter check_token_pass_state.
repeat_pass_token	Same action as pass_token substate.
who_follows	Send who_follows frame. Enter await_response state.
repeat_who_follows_	Same action as who_follows substate.
solicit_any	Send solicit_ successor_2 frame with DA = TS, opening 2 response windows that span all other stations. Enter await_response state.
total_failure	Set sole_active_station true and either silently pass the token back to itself (if the station has more data to send) or enter the idle state. This station will not transmit again unless it has data to send or it hears a valid frame from another station.

5C.2 Access control machine formal description. The access control machine (ACM) is described formally in Reference 4, Appendix I-A.

5C.2.1 List of unique ACM words. This is a list of unique words appearing in the MAC ACM state transition tables of Reference 4, Appendix I-A. This list includes only words from the EXIT CONDITION and ACTION TAKEN parts of these tables; CURRENT STATE, transition name, NEXT STATE and comments (strings beginning with_) were excluded.

```
access_class
any_send_pending
bus_idle_timer
bus_idle_timer.expired
bus_idle_timer.start
bus_quiet
cdu
claim_data_uni
claim_pass_count
claim_timer
claim_timer.expired
claim_timer.start
claim_token
collision
contend_pass_count
contention_delay
contention_timer
contention_timer.expired
contention_timer.start
DA
data_unit
destination
duplicate_address
faulty_transmitter
FC
FCS_suppression
first_time
frame_control
get_pending_frame
heard
hi_pri_token_hold_time
in_ring
in_ring_desired
inter_solicit_count
just_had_token
lowest_address
max_access_class
max_bus_idle
max_transmitter_fault_count
maximum_retry_limit
max_pass_count
max_inter_solicit_count
MA_FAULT_REPORT.indication
MA_INITIALIZE_PROTOCOL.request
noise_burst
nothing
NS
NS_known
octet_time
pass_state
pass_token
power_ok
ring_maintenance_timer
remaining_retries
token_pass_timer.expired
token_pass_timer.start
token_rotation_timer
token_pass_substate'succ
total_failure
transmitter_fault_count
TS
who_follows
```

5D. Frame formats

This part defines the required MAC frame formats. This includes all allowed frame formats and the arrangement of all frame subfields. The term frame as used here refers to the protocol_data_units exchanged by MAC sublayer entities. The MAC_service_data_units received from the PLC sublayer are contained within these MAC frames.

NOTE: Certain frame types exchanged with an alternate Link Control entity (LLC) are shown for completeness. This standard does not specify the formats or use of frames exchanged with LLC entities.

This part describes the frame components and formats used by medium access control. The MAC level transmit frames and abort sequences are described in the following clauses. First the components of the frames are discussed, followed by the definition of the valid frame formats. All frames sent or received by the MAC sublayer shall conform to the following general format:

PREAMBLE	SD	FC	DA	SA	DATA_UNIT...	FCS	ED

```
where PREAMBLE = pattern sent to set receiver's modem clock and
                  level
                  (1 or more octets)
             SD = start delimiter (1 octet)
             FC = frame control (1 octet)
             DA = destination address (2 octets)
             SA = source address (2 or 6 octets)
      DATA_UNIT = information (O or more octets)
            FCS = frame check sequence (4 octets)
             ED = end delimiter (1 octet)
```

The number of octets between SD and ED, exclusive, shall be 1023 or fewer. The abort sequence shall conform to the following format:

SD	ED

```
where SD = start delimiter (1 octet)
      ED = end delimiter (1 octet)
```

Within this part the following acronyms are used for the addresses of the station under discussion, its successor and its predecessor in the logical ring:

TS — this station's address.
NS — next station's address.
PS — previous station's address.

5D.1 Frame components. This clause describes the frame components which are shown in the previous illustrations in greater detail.

5D.1.1 Preamble. The preamble pattern precedes every transmitted frame. Preamble is sent by MAC as an appropriate number of pad_idle symbols. Preamble may be decoded by the receiver as arbitrary data symbols that occur outside frame delimiters. Preamble is primarily used by the receiving modem to acquire signal level and phase lock by using a known pattern. The preamble pattern is chosen for each modulation scheme and data rate for this purpose. Consult the parts on the physical layer for details.

A secondary purpose for the preamble is to guarantee a minimum ED to SD time period to allow stations to process the frame previously received. The minimum amount of preamble transmitted is a function of the data rate. This standard requires that the duration of the preamble shall be at least 2 microseconds, regardless of data rate, and that an integer number of octets shall be sent. At a data rate of 1 Mb/s one octet of preamble is required to meet the integer number of octets requirement. The maximum amount of preamble is constrained by the "jabber" control in the Physical layer. Additionally, for claim token frames, all stations shall use the minimum number of preamble octets to ensure that all frames are of uniform specified length.

5D.1.2 Start delimiter. The frame structure requires a start delimiter, which begins the frame. The start delimiter consists of signaling patterns that are always distinguishable from data.

The start delimiter is coded as follows (see Part 8 for representations of the symbol coding as present on the medium):

```
                          ┌──── First MAC-symbol transmitted
                          ▼
Start Delimiter (SD):    |N N 0 N N 0 0 0|
                          1 2 3 4 5 6 7 8  ◄──── bit positions

 where  N  = non_data MAC-symbol
        0  = zero MAC-symbol
```

5D.1.3 Frame control field. The frame control octet (FC) determines what class of frame is being sent among the following general categories:

1) MAC control
2) LLC data
3) Station management data
4) PLC data

5D.1.3.1 MAC_control_frame

```
 ┌──── First MAC-symbol transmitted
 ▼
|0 0| C C C C C C|
 1 2  3 4 5 6 7 8  ◄──── bit positions
```

where CCCCCC = type of MAC_control frame as follows:

C C C C C C		
3 4 5 6 7 8	bit positions	
0 0 0 0 0 0	Claim_token	
0 0 0 0 0 1	Solicit_successor_1	(has 1 response window)
0 0 0 0 1 0	Solicit_successor_2	(has 2 response windows)
0 0 0 0 1 1	Who_follows	(has 3 response windows)
0 0 0 1 0 0	Resolve_contention	(has 4 response windows)
0 0 1 0 0 0	Token	
0 0 1 1 0 0	Set_successor	

5D.1.3.2 Data_frames

```
    ┌────── First MAC-symbol transmitted
    ▼
┌──────┬──────────┬─────────┐
│ F F  │  M M M   │  P P P  │
└──────┴──────────┴─────────┘

 1  2    3  4   5    6  7  8


 F F = Frame_type:
       (1 2    bit position)

        0 1 = LLC_data_frame
        1 0 = Station_management_data_frame
        1 1 = PLC_data_frame

M M M = MAC_confirmation_class:

       (3 4 5    bit positions)

        0 0 0 = Request_with_no_response
        0 0 1 = Request_with_response
        0 1 0 = Response

P P P = priority:

       (6 7 8  bit positions)

        1 1 X = highest priority
        1 0 X = second highest priority
        0 1 X = third highest priority
        0 0 X = lowest priority
            ▲
            └──── least significant bit of priority class

 X = value has no significance to MAC but may contain information
     of significance to higher layers
```

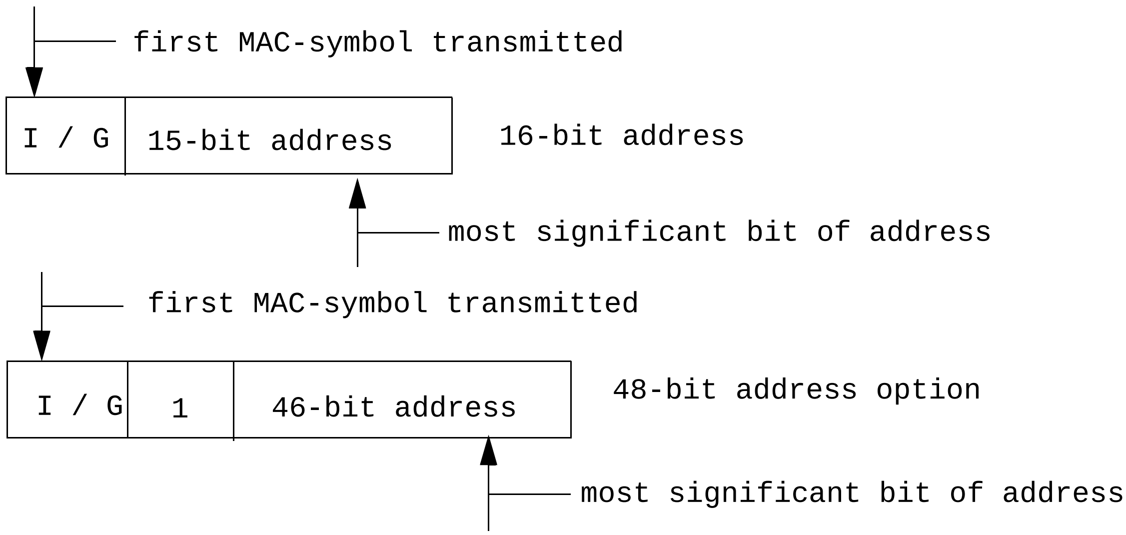
Other bit patterns in the frame control octet are reserved for future study. The action of a station upon receiving an FC value not defined in this standard is not specified.

5D.1.4 Address fields. Each frame shall contain two address fields: the destination address field and the source address field, in that order. Addresses shall be 16 bits or optionally 48 bits in length.

NOTE: **Appendix 5-B** shows examples of individual PROWAY address assignments.

5D.1.4.1 Destination address field composition. The following illustration shows the possible representations of destination addresses.

5D.1.4.1.1 Generic address form

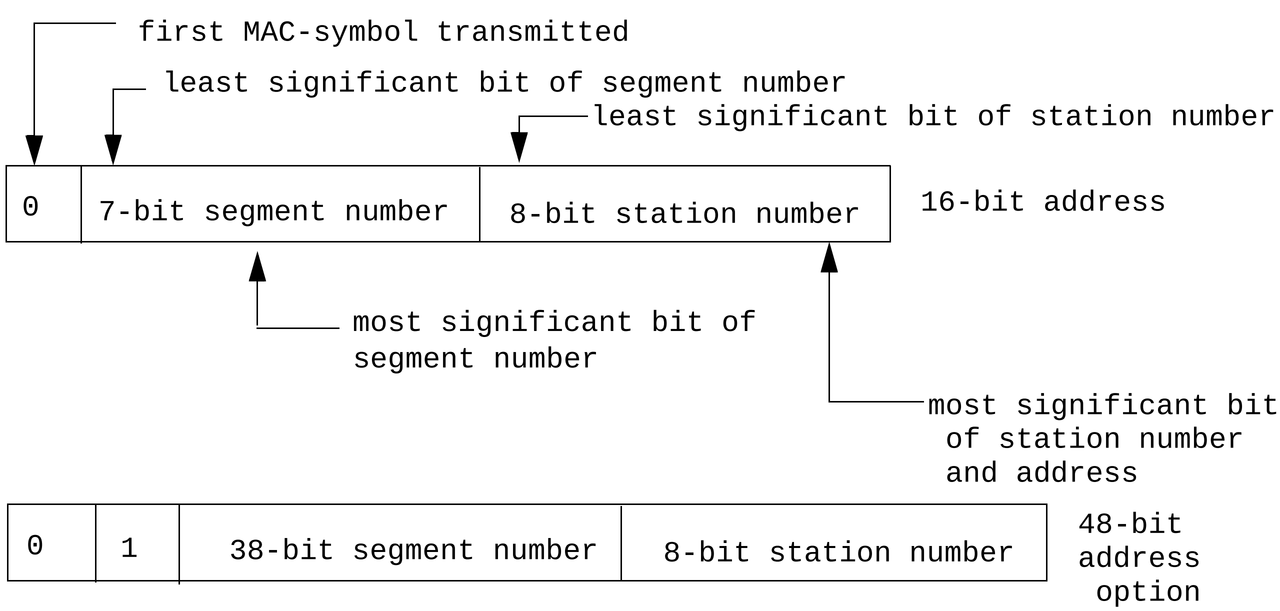


where I/G = Individual/Group address indication bit

The first MAC-symbol transmitted of the destination address (the I/G bit) distinguishes individual addresses from group addresses:

0 = individual address
1 = group address

5D.1.4.1.2 Individual PROWAY station addresses

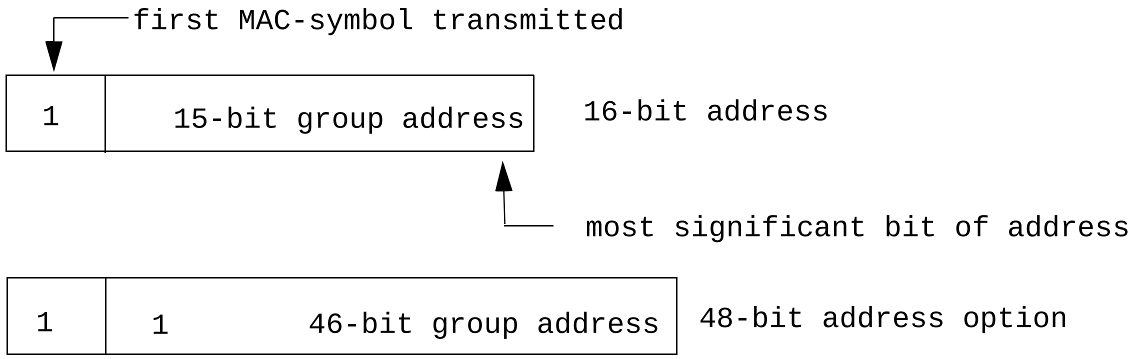


Station number identifies a unique station within a subnetwork, i.e., data highway. Station number may be used for maintenance of the active_station_list.

The lowest allowed individual station number is 0, and the highest is 255. The lowest allowed segment number is 0, and the highest is 127 for 16-bit address and 2^{38} —1 for 48-bit addresses.

Segment number identifies the particular subnetwork within a LAN on which this station resides (see Appendix 5-A). When a LAN contains no subnetworks, it is recommended that segment number = 0.

5D.1.4.1.3 Group addresses



5D.1.4.2 Individual addresses. An individual address identifies a particular station on the LAN and shall be distinct from all other individual station addresses on the same LAN. It consists of the identifying station number assigned to that station by the local administrative authority, preceded by an I/G bit = 0, a "0" (for 48-bit addresses), and a segment number assigned by the local administrative authority.

Group addresses. A group address is used to address a frame to multiple destination stations. Group addresses may be associated with zero, one, or more stations on a given LAN. In particular, a group address is an address associated by convention with a group of logically related stations.

A group address shall not be used in a request_with_response data frame.

Recommended assignments of group addresses in PROWAY networks is for future study.

Broadcast addresses: The group address consisting of all <u>ones</u> (that is, 16 <u>ones</u> for 16-bit addressing or 48 ones for 48-bit addressing, respectively) shall constitute the broadcast address, denoting the set of all stations on the given LAN.

A broadcast address shall not be used in a request_with_response data frame.

NOTE: For some of the frame types used by the token bus MAC procedures, the contents of the destination address, field is irrelevant. In such cases, the originating station's own address or any other properly formed address can be sent in this field.

5D.1.4.3 Source address field. The source address identifies the station originating the frame and has the same format and length as the destination address in a given frame, except that the individual/group bit shall be set to 0; the significance of this bit being set to 1 is a subject for future study.

5D.1.4.4 Numerical interpretation of addresses. Strictly speaking, addresses are bit strings which serve as unique station identifiers or group identifiers. For the purpose of the MAC address comparison within the token bus MAC sublayer, as used in ordering the logical ring and as expressed in the formal access control machine of 5C, each MAC_address bit string is interpreted as if it were an unsigned integer value sent least significant bit first, and thus as if the last bit transmitted had the highest numeric significance.

Additionally the address bits are used in determining delays in the contention process and transmission lengths in the token claiming process. These processes start with the most significant address bits using two bits at a time. Thus the internal processing order is from the serial transmission order on the medium.

5D.1.5 MAC data unit field. Depending on the bit pattern specified in the frames's frame control octet, the MAC data unit field can contain either of the following:

1) A PLC protocol data unit, as specified in Part 3, which is used to exchange PLC information between PLC entities.

2) An LLC protocol data unit, which is used to exchange LLC information between LLC entities. This standard does not specify the details of exchanges between LLC entities.

3) A MAC management data frame, which is used to exchange MAC management information between MAC management entities.

4) A value specific to one of the MAC control frames.

Each octet of the MAC data unit field shall be transmitted low-order bit first.

The question of exchanging PLC data units using the LLC data unit type is for further study and harmonization with the IEEE 802 committee.

5D.1.6 Frame Check Sequence (FCS) field. The FCS is a 32-bit frame checking sequence, based upon the following standard generator polynomial of degree 32:

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^{8} + X^{7} + X^{5} + X^{4} + X^{2} + X + 1$$

The FCS is the one's complement of the sum (modulo 2) of:

1) The remainder of

   $$X^{K} * (X^{31} + X^{30} + X^{29} +... + X^{2} + X + 1)$$

   divided (modulo 2) by the standard 32-bit generating polynomial, where K is the number of bits in the frame control, address (SA and DA), and MAC data_unit fields, and

2) The remainder of the division (modulo 2) by the standard generating polynomial of the product of X^{32} by the content of the frame control, address (SA and DA), and MAC data_unit fields. The FCS is transmitted commencing with the coefficient of the highest degree term.

As a typical implementation, at the transmitter, the initial content of the register of the device computer the remainder of the division is preset to all ones and is then modified during division of the frame format, address and information fields by the generator polynomial (as described above). The one's complement of the resulting remainder is transmitted as the 32-bit FCS sequence.

At the receiver, the initial content of the register of the device computing remainder is preset to all ones. The serial incoming protected bits and the FCS, when divided by the generator polynomial, results, in the absence of transmission errors, in a unique non-zero remainder value. The unique remainder value for the 32-bit FCS is the polynomial:

$$X^{31} + X^{30} + X^{26} + X^{25} + X^{24} + X^{18} + X^{15} + X^{14} + X^{12} + X^{11} + X^{10} + X^{8} + X^{6} + X^{5} + X^{4} + X^{3} + X + 1$$

NOTE: In order to test the FCS generation and checking logic in a station, an implementation should provide a means of bypassing the FCS generation circuitry and providing an FCS from an external source. The ability to pass frames that have FCS errors along with the received FCS value and an error indication to higher levels of the protocol is another desirable testability feature.

5D.1.7 End delimiter. The frame structure requires an end delimiter (ED), which ends the frame and determines the position of the frame check sequence. The data between the SD and ED must be an integral number of octets. All bits between the start and end delimiters are covered by the frame check sequence.

The end delimiter consists of signaling patterns that are always distinguishable from data. The end delimiter also contains bits of information that are not error checked. The end delimiter is coded as follows:

```
                        First MAC-symbol transmitted

End Delimiter (ED):    | N N 1 N N 1 I R |

                         1 2 3 4 5 6 7 8  <---- bit positions

 where N = non_data MAC-symbol
       1 = one MAC-symbol
       R = reserved and set = 0. Non-PROWAY stations may set this bit equal
           one if they detect an error in this frame.
       I = intermediate bit
```

The seventh ED MAC-symbol is called the **intermediate** bit. If **one**, it indicates that more transmissions from the station follow. If **zero**, it indicates that this is the last frame transmitted by the station and silence follows the ED. The I bit will be **zero** for request_with_response and response frames.

5D.1.8 Abort sequence. This pattern prematurely terminates the transmission of a frame. The abort sequence is sent by a station that does not wish to continue to send a frame it has already begun.

```
   First MAC-symbol transmitted

| N N 0 N N 0 0 0 | N N 1 N N 1 I R |

        SD                 ED
```

```
where N = non_data MAC-symbol
      0 = zero MAC-symbol
      l = one MAC-symbol
      R = reserved
      I = intermediate
```

5D.2 Enumeration of frame types. This clause shows how the components of the frames are arranged in the various frame types transmitted by the MAC sublayer. Part 3 discusses the frames and terminology used here.

5D.2.1 MAC control frame formats. The following frames are sent and received by the MAC sublayer and are not passed to higher layers.

5D.2.1.1 Claim_token. The frame has a data_unit whose value is arbitrary and whose length in octets (between addresses and FCS exclusive) is 0, 2, 4, or 6 times the system's slot_time also measured in octets.

PREAMBLE	SD	0 0 0 0 0 0 0 0	DA	SA	arbitrary value, length = (0,2,4,6) slot_time octets	FCS	ED

5D.2.1.2 Solicit_successor_1. The frame has a DA = the contents of the station's NS register and a null data unit. One response window always follows this frame.

PREAMBLE	SD	0 0 0 0 0 0 0 1	DA	SA	FCS	ED	.....

one response window

5D.2.1.3 Solicit_successor_2. The frame has DA = the contents of the station's NS or TS register and a null data unit. Two response windows always follow this frame.

PREAMBLE	SD	0 0 0 0 0 0 1 0	DA	SA	FCS	ED	....

two response windows

5D.2.1.4 Who_Follows. The frame has a data_unit = the value of the station's NS register. The format and length of the data_unit are the same as a source address. Three response windows always follow this frame. (This gives receivers two extra slot_times to make a comparison with an address other than TS.)

PREAMBLE	SD	0 0 0 0 0 0 1 1	DA	SA	value NS	FCS	ED	....

three response windows

5D.2.1.5 Resolve_contention. The frame has a null data_unit. Four response windows always follow this frame.

PREAMBLE	SD	0 0 0 0 0 1 0 0	DA	SA	FCS	ED	.... four response windows

5D.2.1.6 Token. The frame has DA = the contents of the station's NS register and has a null data_unit.

PREAMBLE	SD	0 0 0 0 1 0 0 0	DA	SA	FCS	ED

5D.2.1.7 Set_successor. The frame has DA = the SA of the last frame received, and data_unit = the value of the station's NS or TS register. The format and length of the data_unit are the same as that of a source address.

PREAMBLE	SD	0 0 0 0 1 1 0 0	DA	SA	new value of NS	FCS	ED

5D.2.2 Data frame formats. Data frames have a DA and data_unit specified by the source station's link control or station management entity (as indicated by the FF field). Valid data frames with non-null data_units are delivered to the destination link control or station management entity specified by the FF field by the destination MAC entity.

There are three possible types of data frames.

5D.2.2.1 Request_with_no_response data frame. For a request_with_no_ response data frame, the destination MAC entity does not respond immediately to the frame.

PREAMBLE	SD	F F 0 0 0 P P P	DA	SA	Data_unit	FCS	ED

5D.2.2.2 Request_with_response data frame. For a request_with_response data frame, the destination (responding) MAC entity passes the data unit to the destination PLC entity. The destination PLC entity immediately returns a response data_unit to the destination MAC entity, and the destination MAC entity immediately sends the corresponding response frame.

A station shall not send a request_with_response frame using a group or broadest destination address.

Conceptually a request_with_response frame delegates to the recipient the right to transmit a single frame addressed to the token holder.

PREAMBLE	SD	F F 0 0 1 P P P	DA	SA	Data_unit	FCS	ED

5D.2.2.3 Response data frame. A response data frame is sent by a responding station to the originating (requesting) station upon receipt of a request_with_response frame from the requesting station.

Upon receipt of the response data frame at the requesting station, the originating MAC entity passes the data_unit of the response frame to the originating PLC entity.

PREAMBLE	SD	F F 0 1 0 P P P	DA	SA	Data_unit	FCS	ED

5D.2.3 Invalid frames. An invalid frame is defined as one which meets at least one of the following five conditions:

1) It is identified as such by the Physical layer (for example, it contains **non_data** or **bad_signal** symbols).

2) It is not an integral number of octets in length.

3) It does not consist of a start delimiter, one frame control field, two properly formed address fields, one data unit field of appropriate length (dependent on the bit pattern specified in the frame control field), one FCS field, and an end delimiter, in that order.

4) The FCS computation, when applied to all octets between the SD and the ED, fails to yield the unique remainder specified in 5D.1.6.

5) It is recommended that implementations treat a frame with the intermediate bit of the end delimiter equal to **zero** as an invalid frame unless the ED is followed by at least two MAC_symbol_times of silence.

NOTE: That the intermediate bit equals zero for all request_ with _response and response frames.

Implementations may also treat a frame meeting any of the following additional conditions as an invalid frame:

1) The frame control field contains an undefined bit pattern.

2) The reserved bit within the end delimiter of the frame is asserted. This may indicate that a non-PROWAY station has detected an error.

Invalid frames shall be treated as noise. Their existence, as noise bursts, is relevant at some points in the token bus elements of procedure.

Appendix 5-A An example PROWAY implementation at $1*10^6$ bits/sec

Part 5C allows for ranges of parameter values in implementations of token bus media access control methods. This appendix gives one set of values which meet the access time specifications of Part 1.

MAC_station_delay = 2 octet times

PLC_station_delay = 2 octet times

hi_pri_token_hold_time = 64 octet times

Max_AC_4_rotation_time = 6000 octet times

Max_AC_2_rotation_time = 4000 octet times

Max_AC_0_rotation_time = 2000 octet times

maximum_ring_maintenance_timer = 25000 octet times

maximum_retry_limit = 4

transmission_path_delay = 10 microseconds

max_inter_solicit_count = 255

Typical preamble transmitted = 1 or 2 octet times

ring_maintenance_timer_initial_value = zero

maximum_L_pdu size for SSAP's which use access_class = 6 $\leq$ 16 octets

address length = 16 bits

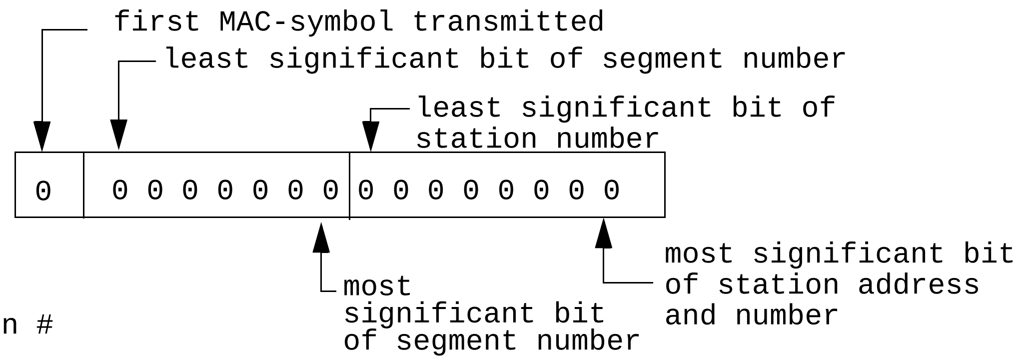
data rate = $1*10^6$ bits/sec

Appendix 5-B Examples of individual PROWAY station addresses

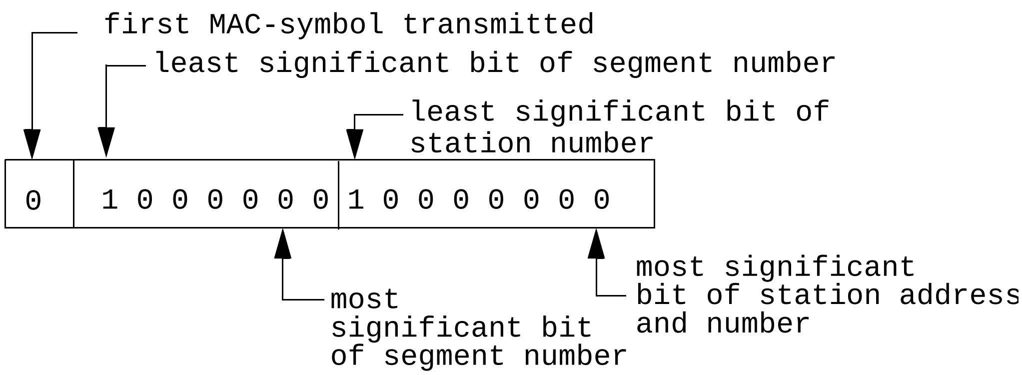
This appendix illustrates the coding of 16-bit station addresses in PROWAY_LAN. It is not a mandatory requirement of this standard.

Segment # / Station #

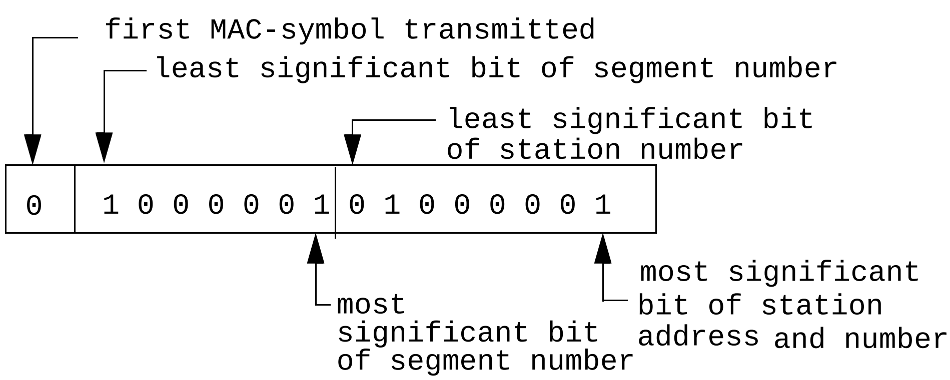
Address Coding



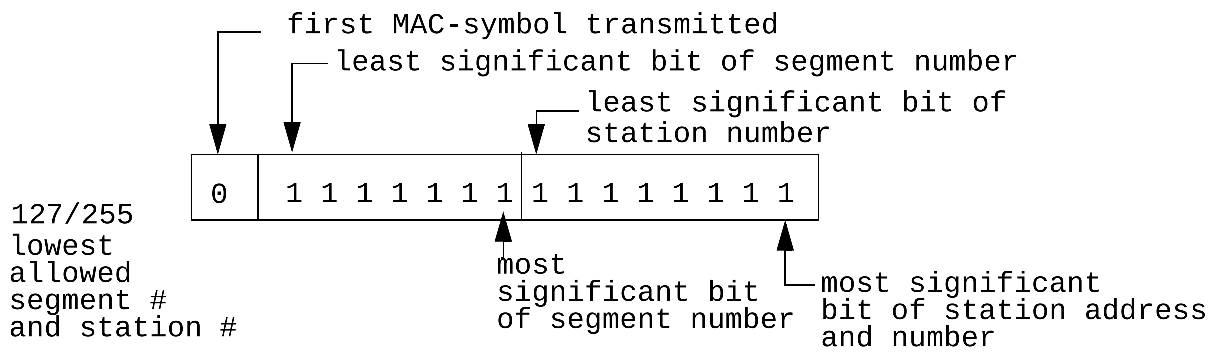
0/0 lowest allowed segment # and station #

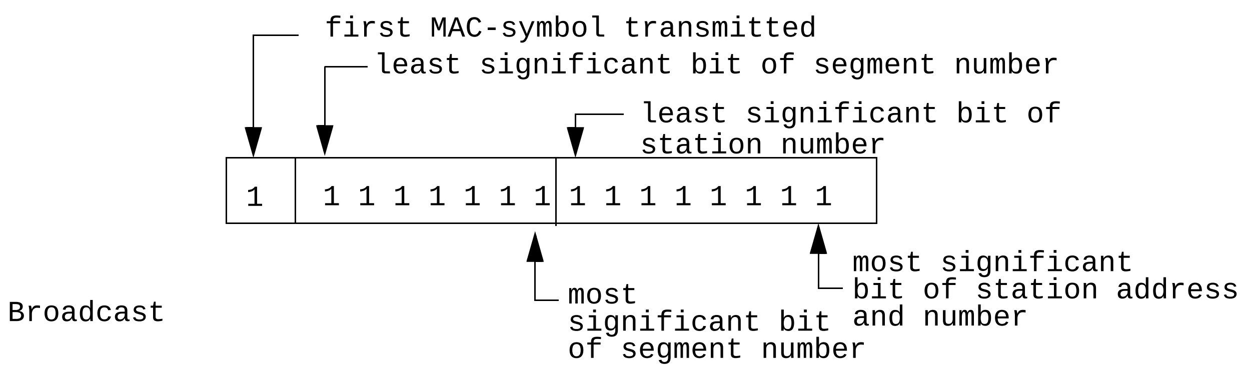


1/1



65/130





An integer representation of a station's address can be calculated using the following formula for 16-bit addresses:

station_address = 2 * (segment_number) + 256 * (station_number)

6 MAC-Physical layer interface specification

This part specifies the interface between the MAC sublayer and the Physical layer.

Part 6A specifies the services at MAC-Physical interface in an abstract way. Part 6A is mandatory for all PROWAY Systems.

Part 6B specifies the implementation of MAC-Physical interface. Part 6B applies only to those stations in which MAC and Physical layers are embodied in two separate pieces of equipment.

The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 6-1.

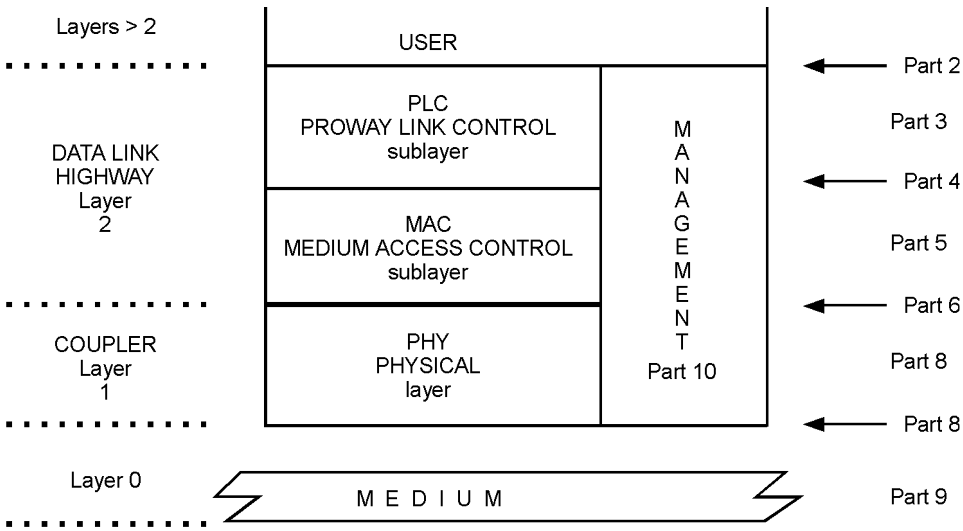


Figure 6-1: Relationship to LAN model

6A. MAC-Physical layer interface service specification

6A.1 Scope and field of application. This part specifies the services provided to the MAC sublayer by the Physical layer of all stations conforming to this standard. It specifies these services in an abstract way. It does not specify or constrain the implementation entities and interfaces within a computer system.

6A.2 Overview of the physical layer service

6A.2.1 General description of services provided by the PHY layer. These paragraphs describe informally the services provided by the Physical layer. These services provide for the transmission and reception of MAC-symbols, each with a duration of one MAC_symbol_ period. Jointly, they provide the means by which cooperating MAC entities can coordinate their transmissions and exchange information by way of a shared communications medium.

6A.2.2 Model used for the service specification. See Reference 1, Appendix I-A.

6A.2.3 Overview of interactions. The primitives associated with symbol transmission and reception are PHY_DATA.request and PHY_DATA.indication.

The **PHY_DATA.request** primitive is passed to the physical layer to request that a symbol be impressed on the local area network's communications medium. Only one such request is accepted per MAC_symbol_period.

The **PHY_DATA.indication** primitive is passed from the physical layer to indicate the reception of a MAC symbol from the medium.

6A.2.4 Basic services and options. All PHY_DATA services are required in all implementations, and both of the PHY_DATA primitives are mandatory.

6A.3 Detailed specifications of interactions with the physical layer entity. This part describes in detail the primitives and parameters associated with the Physical layer services. The parameters specify the information that must be available to the receiving (MAC or Physical) layer entity. A specific implementation is not constrained in the method of making this information available.

6A.3.1 PHY_DATA.request

6A.3.1.1 Function. This primitive is the service request primitive for the MAC symbol transfer service.

6A.3.1.2 Semantics. This primitive shall provide parameters as follows:

PHY_DATA.request (MAC-symbol)

MAC-symbol may specify one of:

1) **zero** — corresponds to a binary 0

2) **one** — corresponds to a binary 1 (**data** is the collective name for **zero** and **one**)

3) **non_data** — used in delimiters, always sent in pairs, and always in octets with the form:
non_data non_data data non_data non_data data data data

4) **pad_idle** — send one symbol of preamble/inter_frame_idle (preamble is a physical-entity sequence of ones and zeros).

5) **silence** — send silence for a duration of one MAC-symbol period. It is defined as the absence of carrier.

6A.3.1.3 When generated. This primitive is passed from the MAC sublayer to the Physical layer to request that the specified symbol be transmitted on the local area network medium. This primitive shall be passed to the Physical layer once for each PHY_DATA.indication that the MAC sublayer receives from the Physical layer. There shall be an implementation-dependent constant phase relationship, determined by MAC, between a PHY_DATA.indication and the next subsequent PHY_DATA.request.

6A.3.1.4 Effect of receipt. Receipt of this primitive causes the Physical layer to attempt to encode and transmit the symbol using the signaling appropriate to the local area network medium. The Physical layer signals its acceptance of the primitive by responding with a locally defined confirmation primitive.

6A.3.1.5 Constraints. pad_idle symbols, which are referred to collectively as **preamble**, are transmitted at the start of each MAC frame, both to provide a training signal for receivers and to provide a non-zero minimum separation between consecutive frames. The following constraints apply:

- An originating station must transmit a minimum number of octet multiples of **pad_idle** such that their duration is at least eight microseconds, and after completing transmission of the last required octet, it may (but need not) transmit more octets of **pad_idle** symbols before the first frame delimiter.

- **non_data** symbols shall be used only within frame delimiters, where they always shall be requested in pairs. The symbol sequences of those frame delimiters shall be: **non_data non_data data non_ data non_data data data data** where each **data** symbol is either the symbol **zero** or the symbol **one**.

- When **data** symbols are transmitted between frame delimiters, the number of **data** symbols transmitted, not including those **data** symbols within the eight-symbol frame delimiter sequences, shall always be a multiple of eight. (That is, only complete octets of data symbols may be transmitted between frame delimiters.) When **pad_idle** symbols are transmitted between frame delimiters, the number of **pad_idle** symbols transmitted shall always be a multiple of eight. Octets of **pad_idle** symbols and octets of **data** symbols shall always be separated by frame delimiter octets, or a sequence of **silence** symbols, or both. (That is, **pad_idle** octets and **data** octets cannot be intermixed.)

- The jitter in the implementation-dependent constant phase relationship between consecutive PHY_DATA.indication and PHY_DATA.request primitives shall not be greater than 2 percent.

6A.3.1.6 Additional comments. The confirmation of this request is a timed confirmation, which can only be made once per transmitted MAC-symbol period. Consequently, this request will only be repeated once per transmitted MAC-symbol period.

6A.3.2 PHY_DATA.indication

6A.3.2.1 Function. This primitive is the service indication primitive for the symbol transfer service.

6A.3.2.2 Semantics. This primitive shall provide parameters as follows:

PHY_DATA.indication (MAC-symbol)

MAC-symbol may specify one of:

1) **zero** — corresponds to a binary 0
2) **one** — corresponds to a binary 1
3) **non-data** — used in delimiters, always sent in pairs
4) **pad_idle** — corresponds to one MAC-symbol period during which preamble/inter_frame_idle was received and reported as a **one**

5) **silence** — corresponds to one MAC-symbol period of received silence (or pseudo-silence)

6) **bad_signal** — corresponds to one MAC-symbol period during which inappropriate signaling was received or implementation-dependent receiver checks fail (refer to Part 8)

6A.3.2.3 When generated. This primitive is passed from the Physical layer to the MAC sublayer to indicate that the specified symbol was received from the local area network medium.

6A.3.2.4 Effect of receipt. The effect of receipt of this primitive by the MAC sublayer entity is defined in Part 5. If a **bad_signal** symbol is received during the reception of a frame, i.e., prior to the receipt by MAC of the end delimiter, MAC will treat this as an error and abort the frame.

6A.3.2.5 Additional comments. This indication is a timed indication, which can only be made once per received MAC-symbol period. Consequently, this indication will only be repeated once per received MAC-symbol period.

Each transmission begins with **pad_idle** symbols, and it is expected that some, but not all, of these initial symbols may be "lost in transit" between the transmitting station and the receiving stations, and consequently reported as **silence**.

Where the Physical layer encoding for successive **pad_idles** is a sequence of **ones** and **zeros**, receivers are permitted to decode such a transmitted sequence of **pad_idles** as a sequence of **ones** and **zeros** and report them as such to the MAC entity. In other words, a receiver need not have the ability to detect and report **pad_idl**e as such; rather it may report the corresponding signaling as **data**.

In the absence of errors or colliding transmissions, and with the above two exceptions for symbols transmitted as **pad_idle**, the sequence of symbols reported is identical to the sequence of symbols transmitted by the associated PHY_DATA.requests.

6B. MAC-Physical layer interface implementation specification

6B.1 Scope and field of application. This part recommends an implementation of MAC-Physical interface. Part 6B applies only to those stations in which the MAC and Physical layers are embodied in two separate pieces of equipment. This part provides a recommended implementation of the abstract primitives found in Part 6A. This part describes these primitives as a set of interface signals, which define the MAC-Physical interface. It specifies the requirements for the interface signals and the interactions between the signals. It also defines the Physical interface requirements including electrical signal levels, grounding technique and connectors. Finally, this part assigns connector pin numbers to the interface signals.

6B.2 Terminology

6B.2.1 Model used for specification. References 11, 12 and 13 of Appendix I-A.

6B.2.2 Compliance. The functions of this section that apply to features implemented in this station are mandatory; however, the open assignments and coding defined are for further study and harmonization with the IEEE 802 committee and ISO TC97/SC6.

6B.3 Overview of interactions

6B.3.1 Signals to the PHY entity. The following signals originate in the MAC entity:

PSC0 Physical_Send_Code_Weight_0
PSC1 Physical_Send_Code_Weight_1
PSC2 Physical_Send_Code_Weight_2
PST Physical_Send_Timing
PLD Physical_Line_Disconnect
PPS Physical_Primary_Signal_Source

6B.3.2 Signals to the MAC entity. The following signals originate in the PHY entity:

PR0 Physical_Receive_Code_Weight_0
PR1 Physical_Receive_Code_Weight_1
PR2 Physical_Receive_Code_Weight_2
PRT Physical_Receive_Timing
PWS Physical_Watchdog_Status

6B.4 Detailed specification of MAC entity-Physical entity interface signals

6B.4.1 Physical_Send signal (PSC0,PSC1,PSC2)

Direction: to PHY entity

The Physical_Send coded signal implements the possible values which may be taken on by the PHY_DATA.request primitive.

6B.4.1.1 Physical_Send encoding. The atomic signals comprising the Physical_Send signal are

PSC0 — Physical_Send_Code_Weight 0

PSC1 — Physical_Send_Code_Weight 1

PSC2 — Physical_Send_Code_Weight 2

The coded value of the Physical_Send signal at the ON to OFF transition of the Physical_Send_Timing (PST) signal defines the value of the PHY_DATA.request primitive for the current MAC_ symbol period according to Table 6-1.

Table 6-1: Physical_Send encoding

PHY.request Value	Physical_Send encoding		
	PSC2	PSC1	PSC0
silence	0	0	X
pad_idle	0	l	X
non_data	1	0	X
zero	1	1	0
one	1	1	1
bad_signal		Not Used	
1 = ON	0 = OFF	X= NOT SIGNIFICANT	

6B.4.1.2 Requirements

1) All of the Physical_Send atomic signals (PSC0, PSC1, PSC2) shall be valid at the ON to OFF transition of the Physical_Send_Timing (PST) signal.

2) The MAC entity shall change the state of the Physical_Send atomic signals (PSC0, PSC1, PSC2) in synchronism with an OFF to ON transition of the Physical_Send_Timing (PST) signal.

3) The open circuit receiver condition of the Physical_Send atomic signals (PSC0, PSC1, PSC2) shall be OFF.

6B.4.2 Physical_Receive signals (PRC0, PRC1, PRC2)

Direction: From PHY entity

the Physical_Receive coded signal implements the possible values which may be taken on by the PHY_DATA.indication primitive.

6B.4.2.1 Physical_Receive encoding. The atomic signal comprising the Physical_Receive signal are

PRC0 — Physical_Receive_Code_Weight_0
PRC1 — Physical_Receive_Code_Weight_1
PRC2 — Physical_Receive_Code_Weight_2

The value of the Physical_Receive signal at the ON to OFF transition of the Physical_Receive_Timing (PRT) signal defines the value of the PHY_DATA.indication primitive for the current MAC-symbol period, according to Table 6-2.

Table 6-2: Physical_Receive encoding

PHY.indication Value	Physical_Receive encoding		
	PRC2	PRC1	PRC0
silence	0	0	X
pad_idle		Not used	
non_data	1	0	X
bad_signal	0	1	X
zero	1	1	0
one	1	1	1
1=ON 0=OFF X=NOT SIGNIFICANT			

6B.4.2.2 Requirements

1) All of the Physical_Receive atomic signals (PRC0, PRC1, PRC2) shall be valid at the ON to OFF transition of the Physical_ Receive_Timing (PRT) signal.

2) The MAC entity shall change the state of the Physical_Receive atomic signals (PRC0, PRC1, PRC2) in synchronism with an OFF to ON transition of the Physical_Receive_Timing (PRT) signal.

3) The open circuit receiver condition of the Physical_Receive atomic signals (PRC0, PRC1, PRC2) shall be OFF.

6B.4.3 Timing signals

6B.4.3.1 Physical_Send_Timing (PST)

Direction: To PHY entity

6B.4.3.1.1 Function. The Physical_Send_Timing (PST) signal establishes the transmission bit rate of the PHY entity and provides the PHY entity with signal element timing information.

6B.4.3.1.2 Requirements

1) The Physical_Send_Timing (PST) signal shall have ON and OFF states for nominally equal periods of time at a frequency corresponding to the bit rate of the MAC entity.

2) The OFF to ON transition of the Physical_Send_Timing (PST) signal shall cause the PHY entity to interpret the current value of the PHY_DATA.request primitive in accordance with 6B.4.1.

3) The open circuit receiver condition of Physical_Send_Timing (PST) shall be OFF.

4) The MAC entity shall maintain a constant phase relationship between the PST andPTT signals.

6B.4.3.2 Physical_Receive_Timing (PRT)

Direction: From PHY entity

6B.4.3.2.1 Function. The Physical_Receive_Timing (PRT) signal provides the MAC entity with signal_element_timing information.

6B.4.3.2.2 Requirements

1) The ON to OFF transitions of Physical_Receive_Timing (PRT) signal shall cause the MAC entity to interpret the current value of the PHY_DATA.indication primitive in accordance with Section 6B.4.3.

2) The open circuit receive condition of the Physical_Receive_Timing (PRT) signal shall be off.

6B.4.3.2.3 Additional comments. The Physical_Receive_Timing (PRT) signal is derived by the PHY entity from state transitions on the transmission line including **preambles**.

6B.4.3.3 Physical_Transmit_Timing (PTT)

Direction: From PHY entity

6B.4.3.3.1 Function. The Physical_Transmit_Timing (PTT) signal provides the MAC entity with accurate timing information and is used to establish signal_element_timing with the MAC entity.

6B.4.3.3.2 Requirements

1) The Physical_Transmit_Timing (PTT) signal is derived by the PHY entity from a frequency source which meets the requirements of 8.5.1.

2) The open circuit receiver condition of the Physical_ Transmit_ Timing (PTT) signal shall be OFF.

3) The PHY entity shall begin to provide timing information on the PTT signal when the power supply to the PHY entity is switched on.

6B.4.4 Management signals

6B.4.4.1 Physical_Line_Disconnect (PLD)

Direction: To PHY entity

6B.4.4.1.1 Function. The Physical_Line_Disconnect (PLD) signal implements the line_disconnect value of PHY_RESET.request primitive. This enables and disables the PHY entity transmitters.

6B.4.4.1.2 Requirements

1) When the Physical_Line_Disconnect (PLD) signal is set to ON, the PHY entity's transmitters will be unconditionally disconnected from the line.

2) When the Physical_Line_Disconnect (PLD) signal is set to OFF, all of the PHY entity's transmitters shall be enabled for normal operation.

3) The ON to OFF transition of the Physical_Line_Disconnect (PLD) signal shall set the watchdog function of its "normal" state, which is actively checking for correct PHY entity performance.

4) This ON to OFF transition of the Physical_Line_Disconnect (PLD) signal shall reset the jabber circuit.

5) The open circuit condition of the Physical_Line_Disconnect (PLD) signal shall be on.

6B.4.4.2 Physical_Watchdog_Status (PWS)

Direction: From PHY entity

6B.4.4.2.1 Function. The Physical_Watchdog_Status (PWS) signal implements the jabber_inhibit value of the PHY_MODE_CHANGE.indication. Thus it indicates the state of the PHY entity watchdog.

6B.4.4.2.2 Requirements

1) The ON condition of the Physical_Watchdog_Status (PWS) signal shall be maintained whenever the PHY entity's watchdog is in the "normal" state.

2) The OFF condition of the Physical_Watchdog_Status (PWS) signal shall indicate that the PHY entity watchdog has detected a fault state.

NOTE: The PWS = OFF condition causes the PHY entity to be disconnect itself from the line.

6B.4.4.3 Physical_Primary_Signal_Source (PPS)

Direction: To PHY entity

6B.4.4.3.1 Function. The Physical_Primary_Signal_Source (PPS) signal directs the PHY entity to receive signals from the primary medium or the alternate medium.

6B.4.4.3.2 Requirements

1) The ON condition of this signal shall cause the primary receiver to be the source of receive signal.

2) The OFF condition shall cause the alternate receiver to be the source of receiving signal.

3) In conjunction with the PLD signal, loopback testing of the primary and alternate receivers and transmitters may be performed. Note that loopback testing requires that the station go offline through the activation of the PLD signal.

4) The open circuit condition of Physical_Primary_Signal_Source (PPS) shall be ON.

6B.5 MAC-PHY interface realization

6B.5.1 Grounding

6B.5.1.1 Signal ground. The signal ground conductor shall connect the MAC entity circuit common (ground) to the PHY entity circuit common (ground) so as to provide a conductive path directly between the two circuit common. See Figure 6-2.

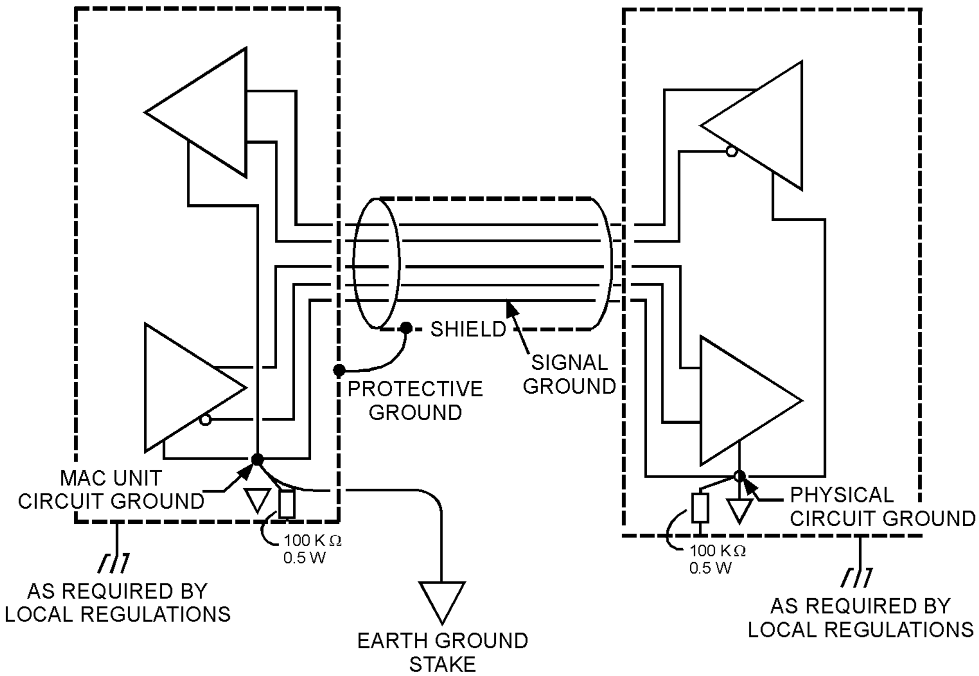


Figure 6-2: MAC-Physical grounding

6B.5.1.2 Protective ground. The protective ground, alternatively termed frame ground, is defined for the purpose of this standard, as the electrical bonding of both the MAC and PHY entity unit to the respective equipment frame. A protective ground shall be provided and the appropriate requirements in Reference 13 of Appendix I-A, shall apply.

6B.5.1.3 External ground connections. Existing National safety or other National regulations shall be observed when any external ground is connected to the system protective ground.

NOTE: For example, some countries may have regulations concerning connections to the "Earth" line of the power supply.

6B.5.1.4 Circuit ground connections. The MAC entity and the PHY entity circuit commons (grounds) should each be connected in series with a resistor to their respective protective ground. When so connected, the nominal value of the resistance shall be 100K ohms with a power rating of not less than 0.5 W. Only the circuit common (ground) of the station shall be connected to real earth ground. An example of grounding arrangements is given in Figure 6-2.

6B.5.1.5 Cable shield connection. The shield of the MAC-PHY interconnecting cable shall be connected to the protective ground. This connection shall be made only at the MAC entity end of the cable by means of a connection between contact 1of the unit's mating connector and protective ground leaving unconnected contact 1 of the mating connector of the opposite unit to minimize susceptibility to and generation of external noise.

WARNING: SIGNIFICANTLY DIFFERENT FRAME POTENTIALS SHOULD BE AVOIDED. THE INTERFACE CONNECTION SYSTEM MAY NOT BE CAPABLE OF HANDLING EXCESSIVE GROUND CURRENTS.

6B.5.2 MAC-PHY interface connector

6B.5.2.1 MAC-PHY interface connector mechanical requirements. The mechanical requirements of the 37-pin interface connector to be used on the MAC and PHY entities, and also at each end of the interface interconnecting cable shall be as specified in Reference 11, Appendix I-A. The MAC and PHY entities shall have fitted to each of them fixed interface connectors which have male contacts and female shells, and each connector shall be equipped with two latching blocks (shown in Figure 6-3), as specified in Reference 11, Appendix I-A. Each end of the interface interconnecting cable shall be fitted with free connectors which have female contacts and male shells. These free connectors shall be equipped with means for latching to the blocks on the fixed connectors.

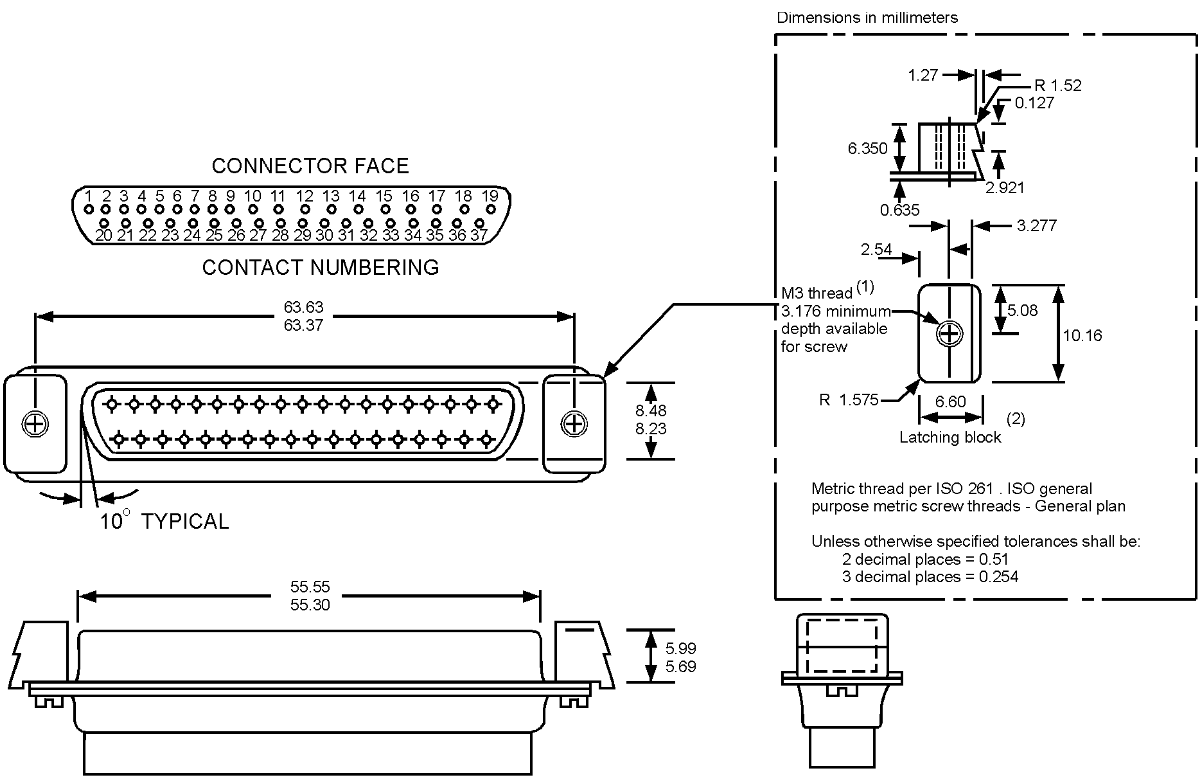


Figure 6-3: MAC-Physical interface connector

NOTE: The means for latching the free connectors to the blocks of the fixed connectors may be subject to National regulations.

6B.5.2.2 Assignment of MAC-PHY connector pin numbers. The pin assignments for the interface signals and other necessary connections are shown in Table 6-3.

Table 6-3: MAC-Physical interface pin assignments

Pin Number	Signal Connection	Pin Number	Signal Connection
1	Shield		
2	Reserved	20	Reserved
3	PSC1	21	PSC1
4	PSC2	22	PSC2
5	PST	23	PST
6	PRC2	24	PRC2
7	PSC0	25	PSC0
8	PRT	26	PRT
9	PWS	27	PWS
10	PLD	28	PLD
11	Reserved	29	Reserved
12	PRC1	30	PRC1
13	PRC0	31	PRC0
14	PPS	32	PPS
15	Reserved	33	Reserved
16	Reserved	34	Reserved
17	PTT	35	PTT
18	Reserved	36	Reserved
19	Signal Ground	37	Reserved

Pins 2-18 inclusive are A-A' interchange points and pins 20-36 inclusive are B-B' interchange points in accordance with Reference 12, Appendix I-A.

The pins designated **Reserved** shall not be used for signal or power connections that have not been defined in this standard.

6B.5.2.3 MAC-PHY interface connector electrical requirements. The MAC-PHY interface connector shall meet the electrical performance requirements as follows:

Voltage rating:	60 V
Testing voltage:	500 V; the voltage tests shall be carried out, as appropriate, in accordance with Clause 9.7.4 of IEC Publication 348
Contact rating:	5 A per contact
Contact resistance:	Less than 20 milliohms
Endurance:	After more than 1000 insertions the contact resistance shall not exceed 20 milliohms
Insulation resistance:	Higher than 5 X 10_E_8 ohms
Contact material:	Gold-plated alloy

6B.5.3 MAC-PHY interconnection cable

6B.5.3.1 Cable length. The maximum length of the MAC-PHY interconnection cable used shall be such that the signal propagation time through this cable does not exceed 20% of nominal MAC_symbol_time as defined in 5B.1. The recommended maximum length is 25 meters at data rate of 1 Mbits/sec.

6B.5.3.2 Cable requirements. The MAC-PHY interconnection cable shall include 17 twisted pairs for the balanced signal interchange circuits and a signal ground conductor. This cable shall embody an electrically conducting shield overall.

6B.5.3.3 Precaution for electromagnetic radiation. In installations prone to excessive electromagnetic radiation, special precautions shall be taken to ensure that the interference EMF induced in the MAC-PHY interconnection cable is brought within acceptable limits.

7 Alternate physical layers and media

Alternate Physical layer and media specifications are given in Chapters 12, 13, 16 and 17 of Reference 4.

8 Specification of the single-channel phase-continuous-FSK Physical (PHY) layer and its interface to the medium

8.1 Scope and field of application. This section specifies the functional, electrical and mechanical characteristics of the Physical (PHY) layer of this standard. This specification defines the Physical layer embodiments found in stations which could attach to a Single-Channel coaxial-cable-bus local area network. The relationship of this section to other sections of this standard and to LAN specifications is illustrated in Figure 8-1.

This standard specifies these Physical layer entities only insofar as necessary to ensure:

1) the interoperability of implementations conforming to this specification, and

2) the protection of the local area network itself and those using it.

8.1.1 Nomenclature. This paragraph defines some terms used in this section whose meanings within the section are more specific than indicated in the glossary of this standard.

Single-channel FSK system: A system whereby information is encoded, frequency modulated onto a carrier, and impressed on the coaxial transmission system. At any point on the medium, only one information signal at a time can be present within the channel without disruption.

Manchester encoding: A means by which separate data and clock signals can be combined into a single, self-synchronizable data stream, suitable for transmission on a serial channel. Within each data bit cell there are always two states, each with a width D/2. If the data bit is a one, the sequence of states low:high is inserted to represent the data value. If the data bit is a zero, the sequence of states high:low is inserted to represent the data value. This method creates a transition in the middle of a bit cell which is retrieved from the signal for use as a clock.

drop cable: A 75 ohm flexible coaxial cable which connects the station to the tap on the trunk cable.

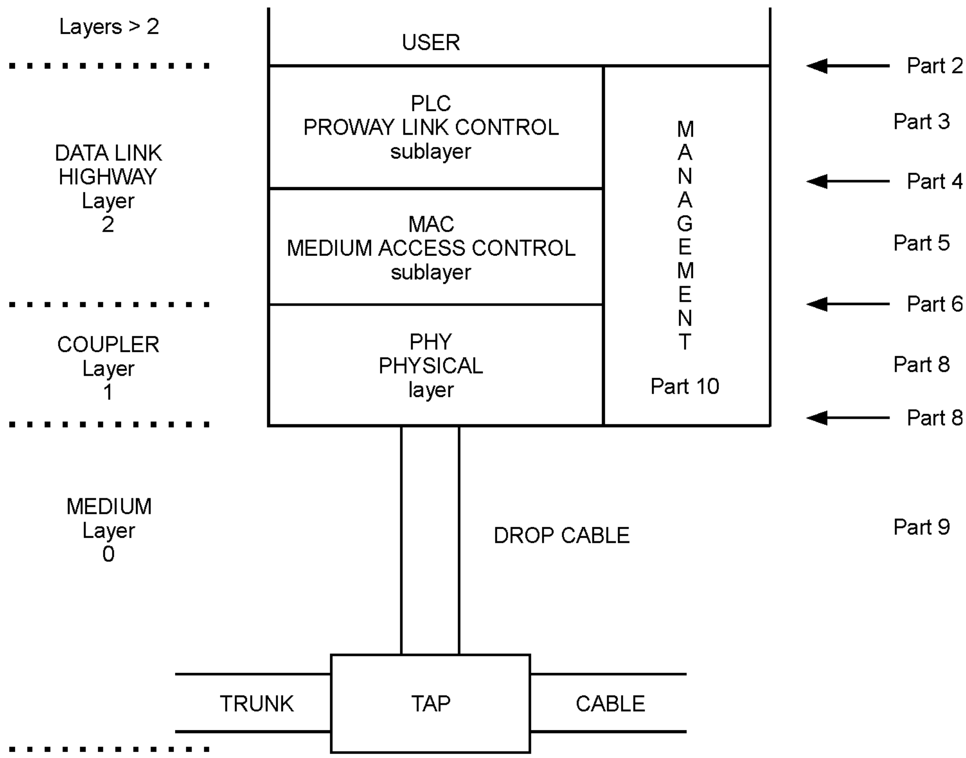


Figure 8-1: Relationship to LAN model and medium

FSK: Frequency shift keying, A modulation technique whereby information is impressed upon a carrier by shifting the frequency of the transmitted signal to one of a small set of frequencies.

phase-continuous FSK: A particular form of FSK where the translations between signaling frequencies is accomplished by a continuous change of frequency (as opposed to the discontinuous replacement of one frequency by another, such as might be accomplished by a switch). Thus it is also a form of frequency modulation.

station: Equipment connected to the local area network.

trunk cable: The main 75 ohm coaxial cable of a single-channel coaxial-cable-bus system.

8.1.2 Object. The object of this specification is to:

1) Provide the physical means necessary for communication between local area network stations conforming to this standard that are connected by a single-channel coaxial-cable-bus medium.

2) Define a physical interface that can be implemented independently among different manufacturers of hardware and achieve compatibility when interconnected by a common single-channel coaxial-cable-bus medium.

3) Provide a communication channel capable of high bandwidth and low bit-error-rate performance. The resultant mean bit-error-rate at the MAC-PHY service interface (see Part 6) shall be less than 1 X 10^{-8}, with a mean undetected-bit-error rate of less than 1 X 10^{-9} at that interface.

4) provide for ease of installation and service in a wide range of environments.

5) provide for high network availability.

8.1.3 Compatibility considerations. This standard applies to Physical layer entities which are designed to operate on a 75 ohm coaxial-cable-bus configured in a trunk and drop cable structure. All single-channel phase-continuous-FSK stations that signal at the same data rate shall be compatible at the medium interface specified in this part. Specific implementations based on this standard may be conceived in different ways provided compatibility at the medium interface is maintained.

8.1.4 Operational overview of the single-channel coaxial-cable-bus medium. The communications medium specified in Part 9 consists of a trunk cable and drop cable structure, with branching (splitters) possible in both the trunk and drop cables by non-directional passive impedance-matching networks (**taps**). The drop cables are connected to the stations.

8.2 Overview of the phase-continuous-FSK Physical (PHY) layer

8.2.1 General description of functions. These paragraphs describe informally the functions performed by the single-channel phase-continuous-FSK physical layer entity. Jointly these physical entities provide a means whereby symbols presented at the MAC interface of one Physical layer entity can be conveyed to all of the Physical layer entities on the bus for presentation to their respective MAC interfaces.

8.2.1.1 Symbol transmission and reception functions. Successive symbols presented to the Physical layer entity at its MAC-PHY interface are applied to an encoder which produces as output a three-state PHY-symbol code: {H}, {L}, {$\underline{0}$}.

That output is then applied to a two-tone FSK modulator which represents each transmitted {H} as the higher frequency tone, each {L} as the lower frequency tone. {$\underline{0}$} represents the OFF condition of the transmitter output.

Each receiver is also coupled to the single-channel coaxial-cable-bus medium. It bandpass filters the received signal to reduce received noise, demodulates the filtered signal and then infers the transmitted PHY-symbol from the presence of carrier and the frequency of the received signal. It then decodes that inferred PHY-symbol by an approximate inverse of the encoding process and presents the resultant decoded MAC-symbols at its MAC-PHY interface.

For all MAC-symbols except **pad_idle**, this decoding process is an exact inverse of the encoding process in the absence of errors. The **pad_idle** symbols, which are referred to collectively as **preamble**, are transmitted at the start of each MAC frame, both to provide a training signal for receivers and to provide a non-zero minimum separation between consecutive frames. Since each transmission begins with **pad_idle** symbols, it is expected that some of these initial symbols may be "lost in transit" between the transmitter and receivers. Additionally, in phase-continuous-FSK systems, the encoding for the MAC-symbols **pad_idle** is a sequence of **ones** and **zeros** and receivers are permitted to decode the transmitted representation of **pad_idle** as a sequence of **ones** and **zeros** and report it as such to the MAC entity.

8.2.1.2 Jabber inhibit function. To protect the local area network from most faults in a station, each station contains a jabber-inhibit function. This function serves as a "watchdog" on the transmitter; if the station does not turn off its transmitter after a prolonged time (roughly 0.1 second), then the transmitter output must be automatically disabled for at least the remainder of the transmission.

8.2.1.3 Local administrative functions. These are functions which select various modes of operation. They are activated either manually, or by way of the Physical layer entity's station management interface, or both. They are

1) Enabling or disabling each transmitter output (a redundant medium configuration would have two or more transmitter outputs).

2) Selecting the received signal source: any single medium (if redundant media are present), or any available loopback point.

NOTE: If a loopback point is selected, then all transmitter outputs must be inhibited.

8.2.2 Model used for the functional specification. See Reference 22 of Appendix I-A.

8.2.3 Required functions. All functions and requirements are mandatory in all implementations.

However, the limits appropriate to sections 8.5.2 and 8.5.3 shall be stated by the manufacturer.

8.3 Application of station management-PHY layer interface specification.

Refer to Part 10D.

8.4 Single-channel phase-continuous-FSK Physical layer functional, electrical and mechanical specifications.

Unless otherwise stated, all voltage and power levels specified are in rms and dB (1 mV, 75 ohm) rms, respectively, based on transmissions of arbitrary data patterns.

8.4.1 Data signaling rates. The standard data signaling rate for phase-continuous-FSK systems is 1 Mbits/sec. The permitted tolerance for this signaling rate is $\pm$ 0.01 percent for a transmitting station.

8.4.2 Symbol encoding. The Physical layer entity transmits MAC-symbols presented to it at its MAC-PHY interface by the MAC entity. The possible MAC-symbols are (see 6A.3.1):

zero
one
non_data
pad_idle
silence

The transmission symbols are

{H}
{L}
{0}

These transmission symbols are applied to a modulator for 1/2 of the bit period and transmitted.

The encoding for each of the input MAC-symbols is

silence	— Each **silence** symbol shall be encoded as the sequence {0 - 0}.
pad_idle	— Each pair of **pad_idle** symbols shall be encoded as the sequence {L} {H} {H} {L}.
zero	— Each **zero** symbol shall be encoded as the sequence {H} {L}.
one	— Each **one** symbol shall be encoded as the sequence {L}{H}.

non_data — Non_data symbols are transmitted by the MAC sublayer entity in pairs. Each such pair of consecutive **non_data** symbols shall be encoded as the sequence {L} {L} {H} {H}.

Thus the **start_frame_delimiter** sequence will be encoded as

non_data	non_data	zero	non_data	non_data	zero	zero	zero
{LL},	{HH}	{HL}	{LL}	{HH}	{HL}	{HL}	{HL}

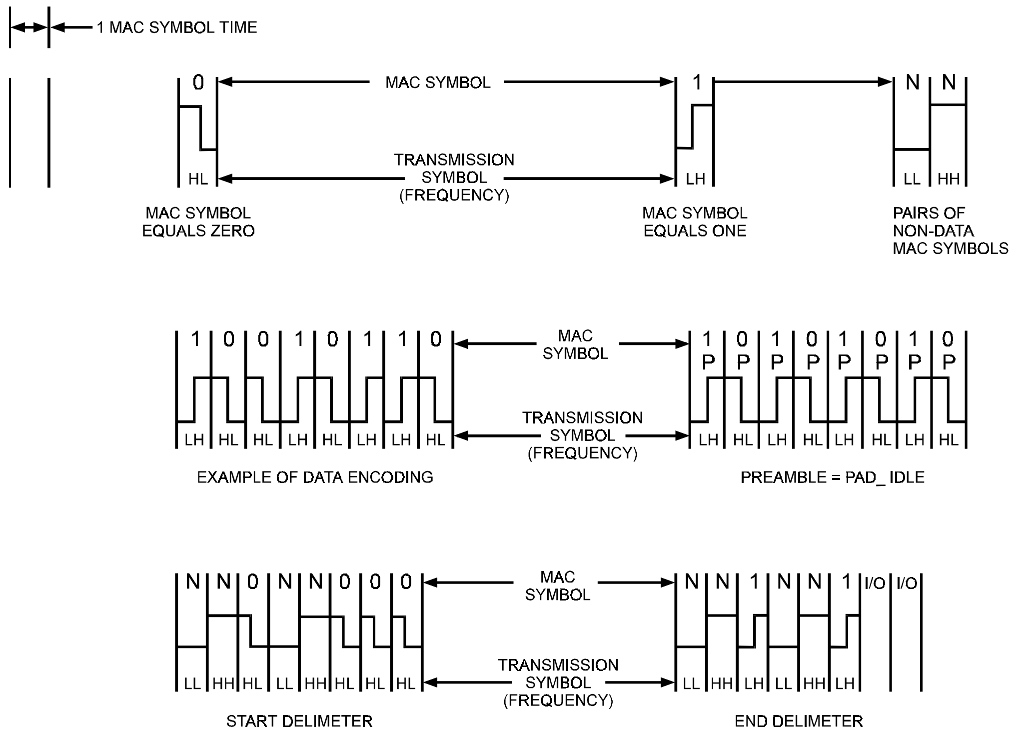


Figure 8-2: Manchester data encoded physical signal encodings

The **end frame delimiter** sequence will be enclosed as

non_data	non_data	one	non_data	non_data	one	one/zero	one/zero
{LL},	{HH},	{LH},	{LL},	{HH},	{LH}		

8.4.3 Modulated line signal (at the line output of the station). The result of the transmission encoding step of 8.4.2 shall be applied to an FSK modulator with the result that each {H} shall be represented by the higher of the modulator's two signaling frequencies, each {L} shall be represented by the lower of the modulator's two signaling frequencies, and each {0} shall be represented by the absence of both carrier and modulation. The resultant modulated carrier shall be coupled to the single-channel coaxial-cable-bus medium as specified in 8.4.5.

8.4.3.1 The line signal shall correspond to an FSK signal with its carrier frequency at 5.00 MHz, varying smoothly between the two signaling frequencies of 3.75 MHz ± 80 kHz and 6.25 MHz ± 80 kHz. **8.4.3.2** Each of the transmission symbols resulting from the transmission encoding step of 8.4.2 shall be transmitted for a period equal to one-half of the inter-arrival time of the MAC-symbols which the MAC entity presents at the MAC interface. The maximum jitter in this periodicity shall be less than one percent of that MAC-symbol inter-arrival time.

8.4.3.3 When transitioning between the two signaling frequencies, the FSK modulator shall change its frequency in a continuous and monotonic manner within 100 ns, with amplitude distortion of at most ten percent.

8.4.3.4 The output level of the transmitted signal at the modulated carrier frequency into a 75 ohm resistive load shall be between +64 and +66 dB (1 mV, 75 ohm) (dBmV).

8.4.3.5 The residual or leakage transmitter-off output signal (that is, while "transmitting" the PHY-code {0}) shall be no more than -22 dB (1 mV, 75 ohm) (dBmV)**.**

8.4.4 Jabber inhibit. Each Physical layer entity shall have a self-interrupt capability to inhibit modulation from reaching the local area network medium. Hardware within the Physical layer shall monitor the output-on condition of the transmitter and shall provide a nominal window of 0.1 second ±25 percent during which time data link transmission may occur. If a transmission is in excess of this duration, the jabber inhibit function shall operate to inhibit any further output from reaching the medium. Reset of this jabber inhibit function shall occur upon receipt of a station management PHY-RESET.request (see Part 10D). Additional resetting means are permitted.

8.4.5 Coupling to the medium. The Physical layer functions are intended to operate satisfactorily over a medium consisting of a 75 ohm bidirectional coaxial trunk cable, nondirectional impedance matching taps, and 75 ohm drop cables. The mechanical coupling of the station to the medium shall be to a drop cable by way of a connector on the station, as specified in Part 9.

The maximum Voltage-Standing-Wave-Ratio (VSWR) at the receiver connector shall be 1.5:1 or less when that connector is terminated with a 75 ohm resistive load. The VSWR is as measured over the spectral range of 3 - 7 MHz.

Both the transmitter and the receiver shall be transformer coupled to the center conductor of one of the medium's drop cables. The breakdown voltage between the windings shall be at least 500 volts ac rms. The shield of the coaxial cable medium may optionally be connected to chassis ground, and the impedance of that connection shall be less than 0.1 ohm.

8.4.6 Receiver sensitivity and selectivity. The Physical layer entity shall be capable of providing an undetected bit error rate of 1×10^{-9} or lower, and a detected bit error rate of 1×10^{-8} or lower, when receiving signals at 4 dBmV to +50 dBmV and with a signal to noise ratio (SNR) of 20 dB or greater. The noise shall be as measured over a spectral range of 3 to 7 MHz as described in 8.4.5.

In addition, each receiver must be able to properly interpret its own station's transmissions.

8.4.7 Symbol timing. Each Physical layer entity shall recover the PHY-symbol timing information contained within the transitions between signaling frequencies of the received signal and shall use this recovered timing information to determine the precise rate at which MAC-symbols should be delivered to the MAC interface. The jitter in this reported MAC-symbol timing relative to the PHY-symbol timing within the received signaling shall be less than eight percent. (When receiving **silence** from the medium, it shall be reported at the MAC interface at the nominal rate determined by 8.4.1 within ±25%).

8.4.8 Symbol decoding. After demodulation and determination of each received PHY-symbol, that PHY-symbol shall be decoded by the process inverse to that described in 8.4.2, and the decoded MAC-symbols shall be reported at the MAC interface. (As noted in 8.2.1.1, receivers are permitted to decode the transmitted representation of **pad_idle** as a sequence of **ones** and **zeros**.)

Whenever a PHY-symbol sequence is received for which the encoding process has no inverse, those PHY-symbols shall be decoded as an appropriate number of **bad_signal** MAC-symbols and reported as such at the MAC-PHY interface. In such cases, the receiving entity should resynchronize the decoding process as rapidly as possible.

8.4.9 Received signal source selection. The ability to select the source of received signaling, either a loopback point within the Physical layer entity or (one of) the (possibly redundant) media, as directed by the station management entity, is required. When the selected source is other than (one of) the media, the PHY entity shall disable transmission to all connected bus media automatically while such selection is in force.

8.4.10 Transmitter enable/disable. The ability to enable and disable the transmission of modulation onto the single-channel bus medium as directed by the station management entity is mandatory.

8.4.11 Redundant media considerations. Embodiments of this standard which can function with redundant media are encouraged, provided that the embodiment **as delivered** will function correctly in a non-redundant single-cable environment. Where redundant media are employed, separate N connectors and jabber-inhibit monitoring shall exist for each medium (although common inhibition is permissible), receiver signal source selection shall be provided capable of selecting any one of the redundant media, and it shall be possible to enable or disable each single transmitter independently of all other redundant transmitters when the source of received signaling is one of the redundant media.

8.4.12 Reliability. The Physical layer entity shall be designed such that its probability of causing a communication failure among other stations connected to the medium is less than 1×10^{-6} per hour of continuous (or discontinuous) operation.

Connectors and other passive components comprising the means of connecting the station to the coaxial cable medium shall be designed to minimize the probability of total network failure.

8.5 Environmental specifications

8.5.1 Safety requirements. This clause sets forth a number of recommendations and guidelines related to safety concerns. The list is incomplete; neither does it address all possible safety concerns. The designer and installer are urged to consult the relevant local, national, and international safety regulations to assure compliance with the appropriate standards. Reference 33 of Appendix I-A provides additional guidance on many relevant regulatory requirements.

Local area network cable systems as described in Part 9 are subject to at least four direct electrical safety hazards during their use, and designers of connecting equipment should be aware of these hazards. The hazards are:

1) direct contact between local network components and power or lighting circuits.
2) static charge buildup on local network cables and components.
3) high-energy transients coupled onto the local network cabling system.
4) potential differences between safety grounds to which various network components are connected.

These electrical safety hazards, to which all similar cabling systems are subject, should be alleviated properly for a local network to perform properly. In addition to provisions for properly handling these faults in an operational system, special measures must be taken to ensure that the intended safety features are not negated when attaching or detaching equipment from the local area network medium of an existing network.

Sound installation practice is defined in Reference 15, Appendix I-A.

8.5.2 Electromagnetic and electric environment. Sources of interference from the environment include electromagnetic fields, electrostatic discharge, transient voltages between earth connections, and so forth. Several sources of interference will contribute to voltage buildup between the coaxial cable and the earth connection, if any, of the station.

The Physical layer entity shall meet its specifications when operating in an industrial environment.

8.5.2.1 Electromagnetic environment. The industrial environment may include an ambient plane wave field of

1) 2 volts/meter from 10 kHz through 30 MHz

2) 5 volts/meter from 30 MHz through 1 GHz

8.5.2.2 Differences in earth potential. Typical differences in earth potential for an industrial environment are:

1) When the medium is entirely contained in a protected area, this difference in earth potential is typically less than 10 V peak-to-peak at frequencies less than 400 Hz.

2) When the medium is exposed to the plant environment, this difference in earth potential is typically less than 50 V peak-to-peak at frequencies less than 400 Hz.

3) When the medium is exposed to a severe plant environment (for example a power station), this difference in earth potential may typically rise to 1000 V peak-to-peak at frequencies less than 10 MHz.

8.5.3 Temperature and humidity. Any embodiment of this standard is expected to operate over a reasonable range of environmental conditions related to temperature, humidity, and physical handling such as shock and vibration. Specific requirements and values for these parameters are considered to be beyond the scope of this standard. Manufacturers are to indicate in the literature associated with system components and equipment (and on the components if possible) the operating environment specifications to facilitate selection, installation and maintenance of these components.

8.5.4 Regulatory requirements. Regulatory requirements that may apply to local area network equipment and media include but may not be limited to those listed as References 15 through 20 and 33 of Appendix I-A.

8.6 Labeling. It is required that each embodiment (and supporting documentation) of a Physical layer entity conformant to this standard be labeled in a manner visible to the user with at least these parameters:

1) data rate capability in Mbits/sec (that is, 1 Mbit/sec).

2) worst-case round-trip delay which this equipment induces on a two-way transmission exchange between stations, as specified in 5B.1.6.

3) operating modes and selection capabilities as defined in 10D.

4) When the station has multiple N-series connectors (for example, for redundant media) the role of each such connector shall be designated clearly by markings on the station in the vicinity of that connector.

9 Single-channel coaxial-cable-bus medium "layer" specification

9.1 Scope and field of application. This section specifies the functional, electrical and mechanical characteristics of the medium "layer" (single-channel coaxial-cable-bus) of the PROWAY standard. This specification defines the medium "layer" embodiment of a single-channel coaxial-cable-bus local area network. The relationship of this section to other sections of this standard and to LAN specifications is illustrated in Figure 9-1.

This standard specifies the medium "layer" only insofar as necessary to ensure:

1) the interoperability of Physical layer entities conforming to this standard when connected to a medium layer conformant to this section, and

2) the protection of the local area network itself and those using it.

9.1.1 Nomenclature

Single-channel FSK system: A system whereby information is encoded, frequency modulated onto a carrier, and impressed on the coaxial transmission medium. At any point on the medium, only one information signal at a time can be present in the channel without disruption.

drop cable: A flexible coaxial cable of the single-channel coaxial-cable-bus medium which connects to a station.

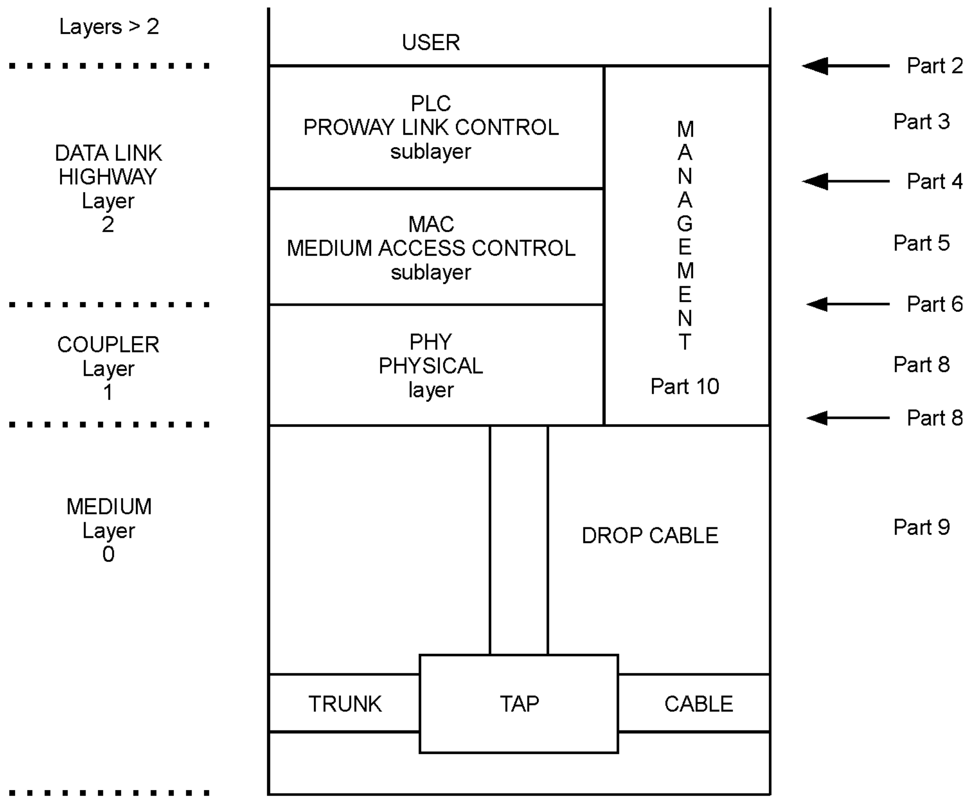


Figure 9-1: Relationship to LAN model

connector: A coaxial connector.

FSK: Frequency shift keying, a modulation technique whereby information is impressed upon a carrier by shifting the frequency of the transmitted signal to one of a small set of frequencies.

(impedance-matching) power splitter: A small module which electrically and mechanically couples one large diameter trunk cable to other large diameter trunk cables, providing a branching topology for each single-channel coaxial cable trunk. A power splitter combines signal energy received at its ports, splitting any signal energy received from a trunk symmetrically among the other trunks. It contains only passive electrical components (R, L, C).

(impedance-matching) splitter: A version of the power splitter used to couple drop cables together symmetrically.

station: Equipment connected to the local area network.

(impedance-matching) tap: A module which electrically and mechanically couples the trunk cable to drop cables. It splits the signal energy received from each trunk cable very asymmetrically, with the bulk of that signal energy passed to the other trunk cable and only a small percentage going to the drop cables. It combines any signal energy received from the drop cables, splits a small part of that signal energy equally among the trunk cables, and passes most of the rest of that combined signal energy back to the drop cables. It contains only passive electrical components (R, L, C).

trunk cable: The main, usually larger diameter, semi-rigid, coaxial cable of a single-channel coaxial cable-bus system.

9.1.2 Object. The object of this specification is to

1) specify the physical medium necessary for communication between stations conforming to this standard,

2) provide for high network availability, and

3) provide for ease of installation and service in a wide range of environments.

9.1.3 Compatibility. All implementations of the medium "layer" conformant to this standard shall be compatible at the interface of the drop cable to the station. Specific implementations based on this standard may be executed in different ways provided compatibility at the actual drop cable to station interface is maintained. The manufacturer shall state the values applicable to Section 9.6.

9.2 Overview of the coaxial-cable-bus medium "layer"

9.2.1 General description of functions. These paragraphs describe informally the functions performed by the single-channel coaxial-cable-bus medium "layer." Jointly these functions provide a means whereby signals presented at the station interfaces to the drop cables can be combined and conveyed to all of the stations connected to any of the medium's drop cables. Thus all stations connected to this medium can communicate.

9.2.1.1 Overview of the single-channel coaxial-cable-bus medium. Stations are connected to the larger diameter **trunk** cable(s) of single-channel coaxial-cable-bus systems by smaller diameter **drop** cables and impedance-matching **taps**. These taps are passive devices which are nondirectional (that is, omni-directional) with regard to signal propagation. The nondirectional characteristics of the tap permits the station's signal to propagate in both directions along the trunk cable. The taps also minimize the effects of reflections due to any impedance mismatches along the trunk or on drop cables.

The topology of the single-channel coaxial-cable-bus system is that of a highly branched tree without a root. The stations are connected as leaves to the tree's branches. Branching is accomplished in the trunk itself by way of **power splitters**, which provide nondirectional coupling of the signals carried on the trunk cables similar to that of the just described taps. Like the taps, the power splitters employ only passive electrical components (R, L, C only).

Branching in the drop cables is provided by (drop cable) splitters which also employ only passive electrical components.

9.2.2 Model used for the functional specification. See Reference 25, Appendix I-A.

9.2.3 Basic characteristics and options. All characteristics are mandatory and required in all medium implementations that claim conformance to this standard. However, the manufacturer shall state the values applicable to Section 9.6.

9.4 Single-channel coaxial-cable-bus medium "layer" functional, electrical and mechanical specifications. The single-channel coaxial-cable-bus medium "layer" is an entity whose sole function (relative to this standard) is signal transport between the stations of a local area network that are connected by the single-channel coaxial-cable-bus medium. Consequently only those characteristics of the medium "layer" which impinge on station-to-station signal transport, or on human and equipment safety, are specified in this standard.

An implementation of the medium "layer" shall be deemed conformant to this standard if it provides the specified signal transport services and characteristics for the stations of a single-channel coaxial-cable-bus local area network, and if it meets the relevant safety and environmental codes.

All measurements specified in the following paragraphs are to be made at the point of station connection to the medium (i.e., drop cable). Unless otherwise stated, all voltage and power levels specified are in rms and dB (1 mV, 75 ohm) [dBmV], respectively, and are based on transmissions of arbitrary data patterns.

9.4.1 Coupling to the station. The connection of the single-channel coaxial-cable-bus medium to the station shall be by way of a flexible 75 ohm drop cable terminated in a male connector.

This combination shall mate with a female connector mounted on the station. The center conductor of the female connector shall be transformer coupled to the station's electronics. The shell of the female connector shall be electrically isolatable from the station by removal of a ground strap.

9.4.2 Characteristic impedance and impedance matching. The characteristic impedance of the single-channel coaxial-cable-bus medium shall be 75 $\pm$ 3 ohm.

The maximum VSWR at each of the medium's N-connectors shall be 1.5:1 or less when the N-connector is terminated with a 75 ohm resistive load. The VSWR shall be measured over a spectral range of 3 to 7 MHz.

9.4.3 Signal level. When receiving the signal of a single station whose transmit level is as specified in 8.4.3.4, the single-channel coaxial-cable-bus medium shall present those signals to each connected station at an amplitude of between +4 and +50 dBmV.

9.4.4 Distortion. The maximum group delay distortion shall be 25 nanosec over the spectral range of 3-7 MHz.

9.4.5 Noise floor and signal-to-noise (S/N) ratio. It is recommended that the in-band (3-7 MHz) noise floor be -15 dB (1 mV, 75 ohm) [dBmV] or less. In all cases the minimum received signal shall exceed the actual noise floor by at least 20 dBmV.

9.4.6 Cable shielding. The shields of all trunk and drop cables shall provide an effective shielding factor of at least 90%.

9.4.7 Compatibility at the station interface. An embodiment of a single-channel coaxial-cable-bus "medium" is deemed to support a specific single-channel coaxial-cable-bus local area network if the requirements of 9.4.1 through 9.4.7 (inclusive) are met when measured from each point of station connection to the medium, independent of which one of the points of station connection is chosen for test signal origination.

9.4.8 Redundancy considerations. As implied by 8.4.11, redundant single-channel coaxial-cable-bus media are encouraged by this standard. Where redundant media are employed, the provisions of 9.4.1 to 9.4.7 shall apply separately and independently to each single non-redundant medium interface.

9.4.9 Reliability. Connectors, taps and other passive components comprising the means of connecting the station to the coaxial cable medium or comprising a part of the medium shall be designed to minimize the probability of total network failure.

9.5 Installation requirements. This clause sets forth a number of recommendations and guidelines related to safety concerns. The list is incomplete; neither does it address all possible safety concerns. The designer is urged to consult the relevant local, national, and international safety regulations to assure compliance with the appropriate standards; see Reference 33, Appendix I-A.

Local area network cable systems are subject to at least four direct electrical safety hazards during their use, and designers of connecting equipment should be aware of these hazards. The hazards are:

1) direct contact between local network components and power or lighting circuits
2) static charge buildup on local network cables and components
3) high-energy transients coupled onto the local network cabling system
4) potential differences between safety grounds to which various network components are connected.

These electrical safety hazards, to which all similar cabling systems are subject, should be alleviated properly for a local area network to perform correctly. In addition to provisions for properly handling these faults in an operational system, special measures must be taken to ensure that the intended safety features are not negated when attaching or detaching equipment from the local area network medium of an existing network.

9.5.1 Sound installation practice. See Reference 15, Appendix I-A. Applicable local codes and regulations shall be followed in every instance in which such practice is applicable.

9.5.2 Grounding

9.5.2.1 The shields of the trunk cable segments and the tap housings of all connected taps shall be connected in series in each branch of a single-channel coaxial-cable-bus medium. This connected conductive path shall be effectively grounded at several points along the length of the trunk cable, and at every point where the cable enters or leaves a building structure. The suggested practice is to effectively ground each tap housing, and to effectively ground the cable shield at least once per hundred meters on long cable runs between tap housings. However where large differences in ground potential exist between grounding points, other practices may be necessary. Effectively grounded means permanently connected to earth through a ground connection of sufficiently low impedance and having sufficient current-carrying capacity to prevent the building up of voltages that may result in undue hazard to connected equipment or to persons.

9.5.2.2 The shields of all drop coaxial cable segments shall be effectively grounded to the tap housings to which they connect.

9.5.2.3 Where there is reason to believe that the ground potential of an exposed shield of a coaxial cable or the housing of a connector or tap differs from the ground potential in the vicinity of that component by more than a few volts, an insulating sleeve or boot or cover should be affixed to that equipment in such a manner as to ensure that users (not installers) of the equipment will not inadvertently complete a circuit between the exposed shield or housing and the local ground through body contact.

9.5.3 Termination of drop cables and taps. All taps and drop cables shall be terminated in a 75 ohm load, except that a small number of drop cables may be temporarily open circuited during maintenance procedures.

9.5.4 Regulatory requirements. The regulatory requirements that may apply to local area network equipment and media include but may not be limited to those given in References 14-20 and 33, Appendix I-A.

In particular, the FCC requirements for radiation from the coaxial-cable-bus medium and connected equipment do apply, and must be considered in any embodiment of the medium "layer" specified in this standard.

9.5.5 Installation and maintenance guidelines. Installation and maintenance guidelines developed within the CATV industry for inter- and intra-facility installation of coaxial cable systems shall be followed where applicable. In addition the following caution shall be observed:

> **CAUTION: At no time should the shield of any portion of the coaxial trunk cable be permitted to float without an effective ground. If a section of floating cable is to be added to an existing cable system, the installer shall take care not to complete the circuit between the shield of the floating cable section and the grounded cable section through body contact.**

9.5.6 Notice. The installation instructions for single-channel coaxial-cable-bus coaxial cable networks and components shall contain language which familiarizes the installer with the cautions and guidelines mentioned in Section 9.5.

9.6 Environmental specifications

9.6.1 Electromagnetic and electric environment. Sources of interference from the environment include electromagnetic fields, electrostatic discharge, transient voltages between earth connections, and so forth. Several sources of interference will contribute to voltage buildup between the coaxial cable and the earth connection, if any, of the station.

The medium "layer" entity embodiment shall meet its specifications when operating in an industrial environment.

9.6.1.1 Electromagnetic fields. The industrial environment may include an ambient plane wave field of:

1) 2 volts/meter from 10 kHz through 30 MHz

2) 5 volts/meter from 30 MHz through 1 GHz

9.6.1.2 Differences in earth potentials. Typical differences in earth potentials for an industrial environment are:

1) typically less than 10 V peak-to-peak at frequencies less than 400 Hz when the medium is entirely contained in a protected area,

2) typically less than 50 V peak-to-peak at frequencies less than 400 Hz when the medium is exposed to the plant environment,

3) typically rise to 1000 V peak-to-peak at frequencies less than 10 MHz when the medium is exposed to a severe plant environment (for example, a power station).

9.6.2 Temperature and humidity. Any embodiment of this standard is expected to operate over a reasonable range of environmental conditions related to temperature, humidity, and physical handling such as shock and vibration. Specific requirements and values for these parameters are considered to be beyond the scope of this standard. Manufacturers are to indicate in the literature associated with system components and equipment (and on the components if possible) the operating environment specifications to facilitate selection, installation and maintenance of these components.

9.7 Transmission path delay considerations. When specifying an embodiment of a medium "layer" which conforms to this standard, a vendor shall state the transmission_path_delay (see 5B.1) as the maximum **one-way** delay which the single-channel coaxial-cable-bus medium could be expected to induce on a transmission from any connected station to any other station. The delays induced by the transmitting and receiving stations themselves should not be included in the transmission_path_delay.

For each potentially worst-case path through the medium, a path delay is computed as the sum of the medium-induced delay in propagating a signal from one station to another. The transmission_ path_delay used for determining the network's slot_time (see 5B.1) shall be the largest of these path delays for the cable system.

These path delay computations shall take into account all circuitry delays in all medium "layer" splitters or other components as well as all signal propagation delays within the cable segments themselves.

The transmission_path_delay shall be expressed in terms of the network's symbol signaling rate on the medium. When not an integral number of signaled symbols, it shall be rounded up to such an integral number. When uncertain of the exact value of the delay, vendors shall state an upper bound for the value.

9.8 Documentation. It is mandatory that each vendor of an embodiment of a medium "layer" entity conformant to this standard provide to the user supporting documentation with at least these parameters:

1) Specific sections of this standard to which the embodiment conforms

2) The transmission_path_delay, as specified in 9.6 and 5B.1

9.9 Network configuration. The medium must provide a transmission path from each station to all other stations with an attenuation of at least 16 decibels (dB) and no more than 60 dB. This is accomplished by attaching each station to the main trunk using a drop cable and tap and calculating two worst-case attenuation values: the first a minimum, generally between the two closest stations; and a second a maximum, often between the two furthest stations. An acceptable design will place both attenuation values within the range given above. Optimum medium performance (i.e., immunity to noise and interference) will occur for designs having minimum attenuation values within the range given.

Since attenuation is dependent on cable length, cable type, and the taps used, it may be necessary to select specific medium components to arrive at an acceptable solution for any given application.

This topic is further discussed in Appendix 9-A.

Appendix 9-A Guidelines for configuring the medium for a single-channel coaxial-cable-bus local area network

This appendix is not a mandatory requirement of this standard. It is included for information only.

The following recommendations for designing and installing local area networks using coaxial cable transmission media are the result of practical experience, and correspond to typical field conditions that are encountered.

9-A.1 Network configuration. The medium must provide a transmission path from each station to all other stations with an attenuation of at least 16 decibels (dB) and no more than 60 dB. This is accomplished by attaching each station to the main trunk using a drop cable and tap and calculating two worst-case attenuation values: the first a minimum, often between the two closest stations; and a second a maximum, generally between the two furthest stations. An acceptable design will place both attenuation values given within the range given above. Optimum medium performance (i.e., immunity to noise and interference) will occur for designs having minimum attenuation values within the range given.

Since attenuation is dependent on cable length, cable type, and the taps used, it may be necessary to select specific medium components to arrive at an acceptable solution for any given application.

9-A.2 Medium components

9-A.2.1 Coaxial cable. Main trunk and drop cables must be selected to meet both the physical and electrical environments and the cable system attenuation objectives. All cables must provide in excess of 90% coverage of the center conductor by the shield. Certain noisy environments may require special cable and shielding considerations.

9-A.2.1.1 Trunk cables. Several example trunk cable types in common use by the CATV industry are given in Table 9-1. Larger diameter cables are normally lower in attenuation but less convenient to install. Cables using lower loss dielectrics are also available but some of these may be more susceptible to physical and environmental abuse. Cables with supporting steel (messenger) wire, armoring, steel tape, flooding compounds, or extra insulation sleeves are available for special applications.

Table 9-1: Example coaxial cable types

Type cable	Typical attenuation dB/100 m at 10 MHz	V max kV (rms)	Typical application
RG-59*	4.8	3.5	Drop cable
RG-6*	3.2	3.5	Trunk or drop cable
RG-11*	1.7	2.0	Trunk cable
0.412	1.1	TBD	Trunk cable
0.500		TBD	Trunk cable
0.750	0.6	TBD	Main trunk

*RG series type but with full shield coverage.

9-A.2.1.2 Drop cables. The drop cables specified in this section are variable lengths of flexible 75 ohm cable, typically not to exceed 30 meters (100') so that loss in the drop cable is less than 1 dB. This length of drop cable permits relative freedom in routing the trunk cable and locating the station. Since the drop cable length is not negligible, the drop cable must be terminated in its characteristic impedance of 75 ohm to preserve the impedance-matched conditions at the tap. Typical drop cable types are shown in Table 9-1.

9-A.2.2 Splitters. The simplest network topology is a long unbranched trunk, requiring the trunk cable to be routed near each station site, in turn. Branched topologies may be implemented by impedance-matched nondirectional splitters, which are three-port passive networks that divide the signal incident at one port into two equal parts that are transmitted to the other two ports. The insertion loss between any two ports of a typical, commercially available nondirectional splitter is 6.1 dB. When branches are implemented by way of such splitters, a separate loss budget must be calculated for each possible end-to-end path, so that the highest loss path can be used to select the trunk cable.

9-A.2.3 Trunk connection (transformer tap). The drop cable is coupled to the trunk cable through a passive, nondirectional, coupling network, the transformer tap, that is impedance-matched at all ports. A fixed small fraction of the signal traveling in either direction on the trunk cable is transferred to the drop cable. A signal originating on the drop cable is attenuated and then propagates out equally in both directions on the trunk. A tap with example attenuation values is shown in Figure 9-2. Taps supporting several drop cables from a single Physical housing and supporting differing trunk to drop cable attenuation values are available. The insertion losses shown represent the way the incident power is divided between the ports; they are not due to dissipation in the network. Typical attenuation values are given in Table 9-2.

In every case, all ports must be matched for proper operation. This means that a 75 ohm termination must be connected to every unused port, and that the cable attached to any port must be properly terminated.

The coupling networks consist only of passive R, L and C elements. Power connections are not required.

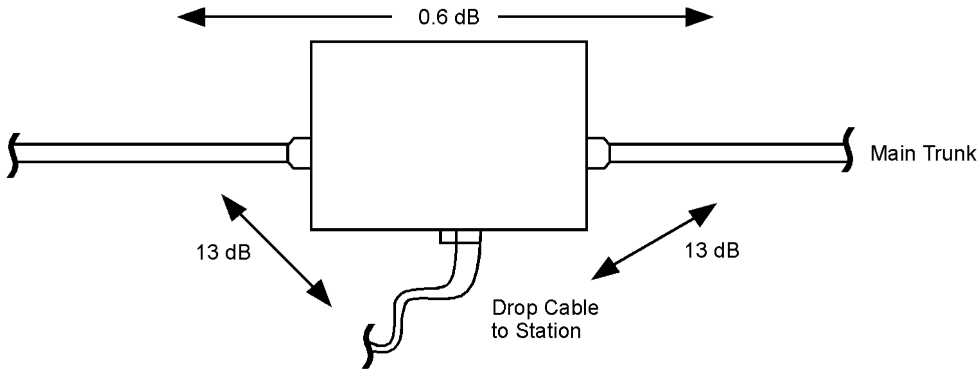


Figure 9-2: Nondirectional transformer tap

Table 9-2: Example specifications of single drop nondirectional transformer taps

Drop Attenuation	Through Attenuation
7 dB	1.8 dB
10	1.0
13	0.6
16	0.4
19	0.3
22	0.2

The coupling networks are enclosed by sealed housings to provide both environmental protection and electrical shielding. Tap housings are available for indoor, outdoor, and below-ground applications. The housings, typically metal castings, include integral connectors for the trunk cable and drop cable(s). The housings for semi-rigid cable are designed such that the coupling network may be removed from the housing without disturbing the trunk connections.

9-A.3 Example network configurations

9-A.3.1 Single control room application. A simple application where only ten PROWAY stations need to be supported over a 100 meter distance is shown in Figure 9-3. An acceptable design using 13 dB single drop taps and RG-11 type coaxial cable is verified by calculating minimum and maximum attenuation values.

Minimum attenuation is between adjacent stations and is 26 dB. Maximum attenuation is between stations 1 and 10 and is 33.3 dB:

Attenuation at tap 1	13
Attenuation of RG-11 main trunk (100 m)	1.5 (@ 7 MHz)
Attenuation of 8 intervening taps	4.8
Attenuation at tap 10	13
Attenuation of cable drop (2 X 40 m)	1.3 (@ 7 MHz)
Total Attenuation	33.6

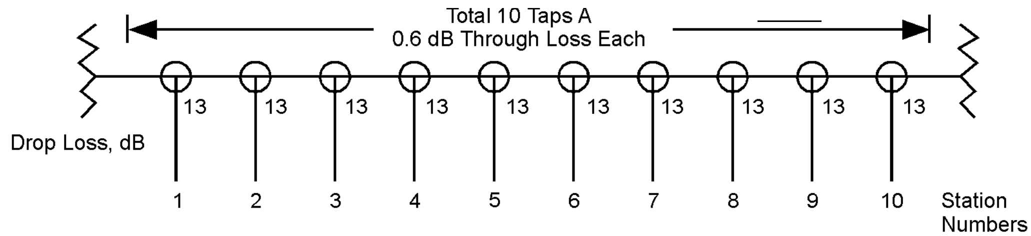


Figure 9-3: Example cable configuration simple control room application

9-A.3.2 Complex plant area application. A complex application where 100 PROWAY stations need to be supported over a 2 km distance is considered. The exact design will depend upon the Physical topology of the application. However, for the purposes of this example it is assumed that half the stations are within 1 km of each other while the remainder are anywhere within the 2 km limit. Each station requires its own tap and up to a 40 meter drop cable between the tap and the station itself.

An initial design using 13 dB taps and 0.412 coaxial cable failed when the maximum attenuation was calculated to be 104.1 dB:

Attenuation at first tap	13
Attenuation of 0.412 main trunk (2 k)	18
Attenuation of 98 taps	58.8
Attenuation of last tap	13
Attenuation of cable drop (2 X 40 m)	1.3
Total Attenuation	104.1

An acceptable design was obtained by using lower attenuation coaxial cable by grading the tap to drop attenuation values as shown in Figure 9-4. The worst-case attenuation for the entire network is within the central 1 km area, which contains 50 taps. Maximum attenuation is between the furthest stations in the core area (stations 26 and 75 in Figure 9-4) and is 58.9 dB:

Attenuation at tap 25	22
Attenuation of 1 km 0.750 main trunk	4
Attenuation of 48 intervening 0.2 taps	9.6
Attenuation of tap 75	22
Attenuation of cable drop (2 X 40 m)	1.3
Total Attenuation	58.9

The worst case for the outlying 50 stations, considering the total 2 km area, occurs when the remaining 1 km of main trunk is arbitrarily placed between stations 25 and 26 or between stations 75 and 76. Assuming this worst-case situation, worst-case attenuation is between stations 1 and 76 and is 57.3 dB:

Attenuation at tap 1	10
Attenuation of 74 intervening taps	19
Attenuation of 2 km main trunk	8
Attenuation at tap 76	19
Attenuation of cable drops (2 X 40 m)	1.3
Total Attenuation	57.3

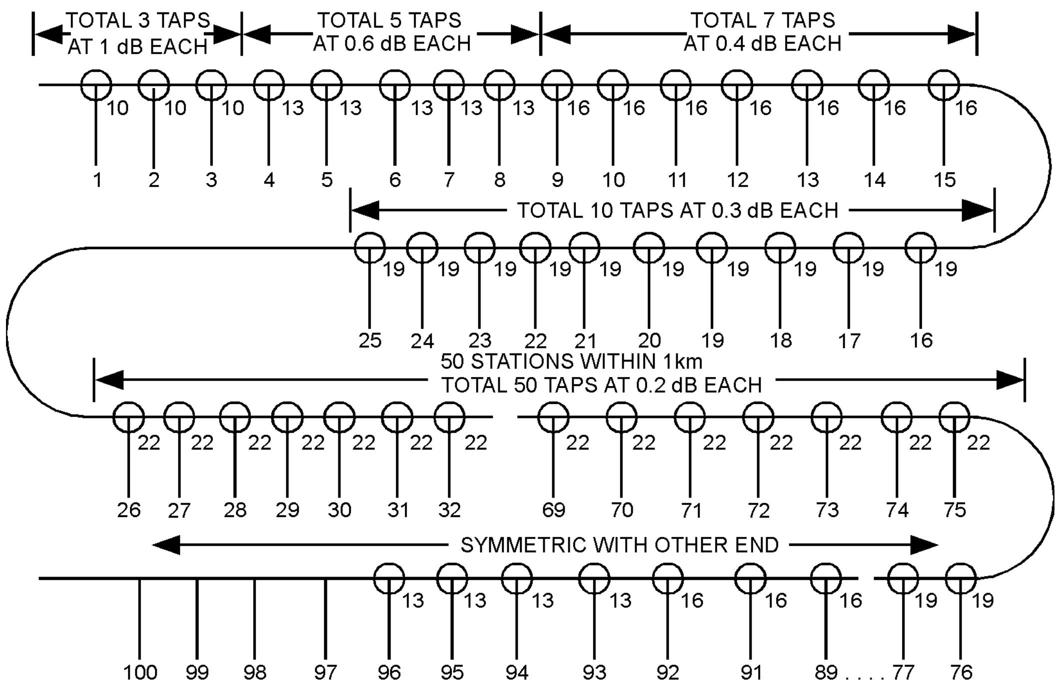


Figure 9-4: Example cable configuration — complex plant area application

The end-to-end attenuation between stations 1 and 100 is 56.1 dB:

Attenuation of taps 1 and 100	20
Attenuation of 98 intervening taps	26.8
Attenuation of 2 km main trunk cable	8
Attenuation of cable drops (2 X 40 m)	1.3
Total Attenuation	56.1

9-A.4 Installation guidelines

9-A.4.1 Grounding. The trunk cable shield may be floating, single-point, or multiple-point grounded as far as signal is concerned. Grounds thus may be installed in compliance with EMI and safety codes and other regulations applicable to the particular installation. This usually means grounding where the cable enters or leaves a building, and at intervals not exceeding approximately 100 meters within the building. Grounds should be applied carefully by a clamp that does not crush or damage the cable, because such cable damage causes serious reflections. Suitable clamps are available from suppliers of CATV system hardware.

9-A.4.2 Surge protection. It is good practice to protect cable against ground surges due to lightning. Suitable protectors that meet the IEEE-472 requirements should be at each end of the cable. The capacitive loading of protectors must be small to avoid affecting physical entity (modem) performance, and must not exceed values for a standard tap. For maximum surge protection, a low impedance, heavy-duty ground connection is required.

9-A.4.3 Termination. The trunk cable must be properly terminated at both ends. All drop cables must be properly terminated at the station end. All unused tap ports must be terminated. Shielded 75 ohm coaxial terminations with broadband characteristics are commercially available for most coaxial cables. Since transmit levels on the trunk cable are approximately 66 dB (1 mV, 75 ohm) [dBmV], power ratings of 1/4 watt are sufficient.

9-A.4.4 Joining cable sections. In general, the trunk cable will consist of a number of separate sections of coaxial cable. Some sections will be joined by connections to tap housings, while others may be joined by splicing connectors (for semi-rigid cable) or straight-through connectors (for flexible cable). Flexible cables will be fitted with matched connectors at each end, while semi-rigid cables will simply have their ends properly prepared to mate with corresponding connectors.

A good engineering practice is to maintain constant impedance between cable sections by using one cable type from one manufacturer for the entire trunk. This practice will avoid significant reflections where cables join. When dissimilar types must be joined, it is suggested that a lossy (attenuating) impedance-matched connector, such as a tap, be employed to reduce reflections.

9-A.4.5 Pretested cable. It is a good practice to pretest all trunk cable before installation. The objectives are to insure that the attenuation does not exceed the expected values at frequencies of interest, and to insure that concealed (that is, internal) discontinuities that can cause reflections do not exist. Most cable suppliers will pretest and certify all cable before shipment for a nominal charge.

On-site testing after installation is also recommended, since any damage may degrade operating margins or cause outright failure. A recommended method for testing the installed cable for damage, improper termination, shorts, or discontinuities is to use a time domain reflectometer, which is available from various instrument manufacturers.

10 PROWAY management

Within the PROWAY architecture, the needs to initiate, terminate and monitor activities within a station, to monitor local area network status, and to recover from abnormal conditions are handled by the management entity of the ISO reference model. Management activities related to layers 1 and 2 of the ISO reference model fall into the following categories:

1) Activation, maintenance and termination of LAN resources at this local station including parameter initialization and modification

2) Monitoring of this local station's status including the reporting of statistics

3) Error control for this local station including the performance of diagnostic, reconfiguration and restart functions

4) Monitoring of participants in the logical token passing ring, i.e., maintenance of the active_station_list

The first three categories are station management activities. They involve exchanges of information between the local user and the local station management entity at the PROWAY management interface (Part 2B) and between the local station management entity and this local station's PLC, MAC and Physical entities at the interfaces defined in Parts10B, 10C and 10D. These local activities are detailed in Part 10A.

The last activity requires communication with management entities of the remote station as well as with the local user and the local MAC entities. The remote communication is accomplished using the MAC_DATA service. Token ring maintenance activities are detailed in Part 10E.

10A. Station management activities

10A.1 Scope and field of application. This part specifies the local station management activities related to layers 1 and 2 of the ISO reference model provided by the management entity of this standard. This standard specifies these activities in an abstract way. It does not specify or constrain the implementation of station management entities and interfaces within a computer system. The relationship of this part to other parts of this standard and to the LAN specifications is illustrated in Figure 10A-1.

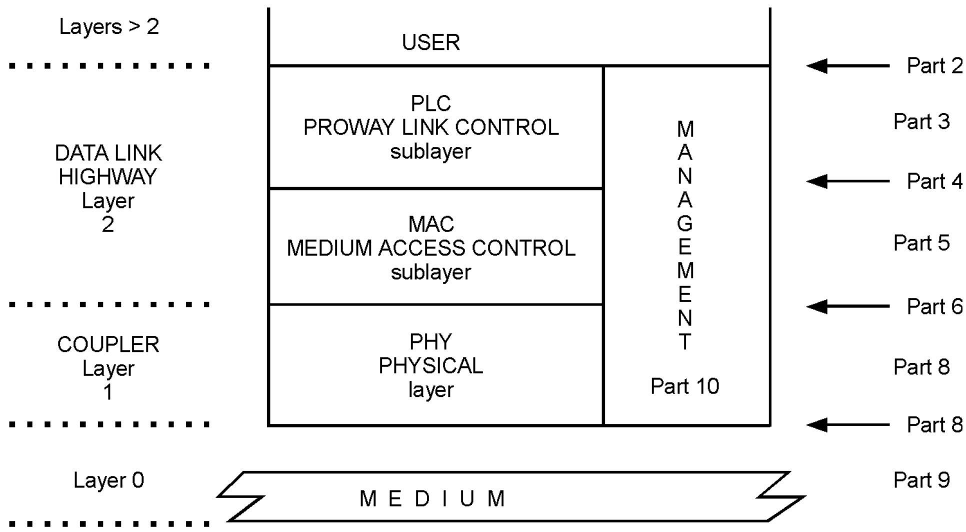


Figure 10A-1: Relationship to LAN model

10A.2 Overview of station management activities.

10A.2.1 User requests. The local user prepares a request for station management and passes this request to the local station management entity over the management interface defined in Part 2B. The local station management entity accepts this service request, performs any specified actions and immediately responds with a confirmation and any requested results. The local station management entity does not exchange information with remote station management entities in satisfying the user requests.

User requests to station management are

- Return Retry Counter
- Enter Line Connect State
- Enter Line Disconnect State
- Enter Configure State
  - Activate LSAP
  - Activate RSAP
  - Deactivate SAP

Activation of the PLC service_access_points (SAPs) is allowed only while the associated PLC state machines are in the Configure state.

Additional user requests to station management are for further study.

10A.2.2 User notification of changes. Important changes in the status of the local station may generate spontaneous indications to the station management entity from the PLC, MAC, or Physical entities. The station management entity conveys this information to a defined local user as an indication.

10A.3 Compliance. The functions of this section that apply to features implemented in this station are mandatory; however, the primitives and coding defined are for further study and harmonization with the IEEE 802 committee.

10A.4 Detailed description of station management activities in response to local user requests.

The local station management entity provides services to the local user in response to user requests.

User requests to station management are made using the L_MGMT.request primitive defined in Part 2B. Station management performs the appropriate actions and responds to the user with an L_MGMT.confirmation, also defined in Part 2B.

The following descriptions detail for each L_MGMT.request the station management actions performed and the conditions which are prerequisite to the action being performed (the states of the management entity referenced are described informally in Appendix 10-B). If the specified conditions are not met, the station management entity shall return a L_MGMT.confirmation with Mgt_ status indicating the cause of failure. The value of results is unspecified in the event of failure. The arguments for each L_ MGMT.request and the results returned with the corresponding L_ MGMT.confirmation are given in Part 2B.

10A.4.1 Return retry counter

Conditions: Local station management entity in Line_Connect state.

Action: Issue MA_READ_VALUE.request (# retries of request_with_ response_frames)

Get MA_READ_VALUE.confirm (# retries of request_with_response_frames)

Results: # retries of request_with_response_frames

10A.4.2 Enter Line_Connect state

Conditions: Local station management entity must be in Line_ Disconnect state.

Action: Local station management entity initializes the PLC, MAC and PHY entities as specified in Parts 10B.3.1,10C.3.3, 10C.3.9, 10D.3.1 and 10D.3.7 and then enters the Line_Connect state. Refer to Appendix 10-B for recommendations of actions the local station management entity should take in transitioning to the Line_Connect state.

No response is returned to the local user until the local station management entity reaches the Line_Connect state.

10A.4.3 Enter Line_Disconnect state

Conditions: Local station management entity must be in Line_Connect state

Action: Local station management entity exits from the token ring gracefully by setting in_ring_desired=**false** and enters the Line_Disconnect state.

Refer to Appendix 10-B for recommendations of actions local station management entity should take in transitioning to the Line_Disconnect state.

No response is made to the local user until the local station management entity reaches the Line_Disconnect state.

10A.4.4 Enter configure state

Conditions: Local station management entity must be in the Line_Connect state with no PLC SAP components activated.

Action: Local station management entity enters the configure state for the purpose of accepting SAP activation requests. This corresponds to entry to the Configure state of the local and remote PLC state machines described in Part 3B.

10A.4.5 Activate L_SAP

Conditions: Local and remote PLC state machines for this SAP must be in the Configure state.

Action: Issue L_SAP_ACTIVATE.request
(SSAP,
Services activated,
Role for each service activated,
Maximum L_sdu_length for each service activated)

Receive L_SAP_ACTIVATE.confirm
(SSAP,
Services_activated,
Role for each service activated,
Maximum L_sdu_length for each service activated)

Local station management entity enters Line_Connect state relative to this SAP when any associated RSAP activation for this SAP is complete. This corresponds to entry to the IDLE and RIDLE states of the local and remote state machines described in Part 3B.

10A.4.6 Activate RSAP

Conditions: Remote PLC state machine for this SAP must be in the configure state.

Action: Issue L_RSAP_ACTIVATE.request
(SSAP to receive L_REPLY.indication,
DSAP associated with shared_buffer_area,
shared_buffer_identification)

Receive L_RSAP_ACTIVATE.confirm
(SSAP to receive L_REPLY.indication,
DSAP associated with shared_buffer_area,
shared_buffer_identification)

Local station management entity enters Line_Connect state relative to this SAP when any activation of this SAP is completed.

10A.4.7 Deactivate SAP

Conditions: Local station management entity must wait until there are no outstanding request_with_response frames for this SAP.

Action: Issue L_SAP_DEACTIVATE.request (SAP),
Receive L_SAP_DEACTIVATED.confirm (SAP).

Local station management entity enters configure state relative to this SAP. This corresponds to entry to the Configure state of the local and remote PLC state machines described in Part 3B.

10A.5 Detailed description of station management activities in notifying the user of changes in station status. The station management entity shall report all

PHY_MODE_CHANGE.indications,
MA_FAULT_REPORT.indications,
MA_EVENT.indications with no successor

to the user identified by SSAP= 01100000 using an L_ MGMT.indication as described in Part 2B. This indication shall identify the cause of the failure as reported by the PHY or MAC indication.

10B. Station Management-PLC sublayer interface service specification

10B.1 Scope and field of application. This part specifies the services provided by the station management entity to the PROWAY link control (PLC) sublayer of this standard. This standard specifies these services in an abstract way. It does not specify or constrain the implementation entities and interfaces within a computer system. The relationship of this to other parts of this standard and to LAN specifications is illustrated in Figure 10B-1.

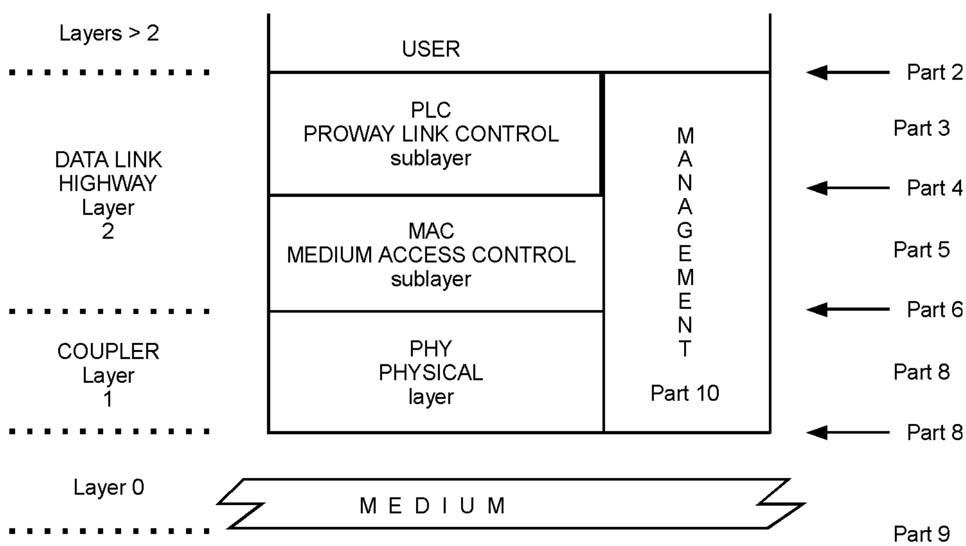


Figure 10B-1: Relationship to LAN model

10B.2 Overview of station management-PLC services

10B.2.1 General description of services provided. This section informally describes the services provided to the station management entity by the PROWAY Link Control (PLC) sublayer. These services are local administrative services between the PLC sublayer and its manager. These services provide the means of:

1) Resetting the PLC entity

2) Activating and configuring the PLCs service_access_points, i.e.,SAPs

3) Specifying the values of constants appropriate for the network

4) Notifying the station management entity of relevant changes in the PLC entity's status.

10B.2.2 Model used for the service specification. Reference 1, Appendix I-A.

10B.2.3 Overview of interactions. The primitives associated with these local administrative services are:

L_RESET.request
L_STATUS.request
L_STATUS.indication
L_STATUS.confirm
L_SAP_ACTIVATE.request
L_SAP_ACTIVATE.confirm
L_RSAP_ACTIVATE.request
L_RSAP_ACTIVATE.confirm
L_SAP_DEACTIVATE.request
L_SAP_DEACTIVATE.confirm

10B.2.3.1 L_RESET. The L_RESET.request primitive is passed to the PLC entity to reset PLC entity.

10B.2.3.2 L_STATUS. The L_STATUS.request primitive is passed to the PLC entity to request a report of the status of a SAP component of a local or remote PLC entity. The L_STATUS.indication primitive is passed from the PLC entity as a result of a remote request for status information or a local change of status. The L_ STATUS.confirm primitive is passed from the PLC entity to convey the results of the previous associated L_STATUS.request.

10B.2.3.3 L_SAP_ACTIVATE. The L_SAP_ACTIVATE.request primitive is passed to the PLC entity to request the activation and configuration of an LSAP component in the PLC ENTITY. This primitive does not activate or configure the RDR response component of this LSAP. The L_SAP_ACTIVATE.confirm primitive is passed from the PLC entity to convey the results of the previous associated L_SAP_ACTIVATE.request.

10B.2.3.4 L_RSAP_ACTIVATE. This primitive is passed to the PLC entity to request the activation and configuration of the component of that SAP that responds to RDR requests. The L_RSAP. confirm is passed from the PLC entity to convey the result of the previously associated L_RSAP_ACTIVATE.request.

10B.2.3.5 L_SAP_DEACTIVATE. The L_SAP_DEACTIVATE.request primitive is passed to the PLC entity to request the deactivation of an SAP component in the PLC entity. The L_SAP_DEACTIVATE. confirm primitive is passed from the PLC entity to convey the results of the previous associated L_SAP_DEACTIVATE.request.

10B.2.3.6 Services to test local and remote PLC entities. For future study are services to configure active SAP's and to test local and remote PLC entities.

10B.2.3.7 Compliance. The functions of this section that apply to features implemented in this station are mandatory; however, the primitives and coding defined are for further study and harmonization with the IEEE 802 committee.

10B.3 Detailed specifications of interactions with the station management entity.

This section describes in detail the primitives and parameters associated with the identified services. Note that the parameters are specified in an abstract sense. The parameters specify the information that must be available to the receiving entity. A specific implementation is not constrained in the method of making this information available.

10B.3.1 L_RESET.request

10B.3.1.1 Function. This primitive is the service request primitive for the PLC sublayer reset service.

10B.3.1.2 Semantics. This primitive shall be parameterless as follows:

L_RESET.request

10B.3.1.3 When generated. This primitive is passed from the station management entity to the PLC sublayer entity to request that the PLC sublayer entity reset itself.

10B.3.1.4 Effect of receipt. Receipt of this primitive causes the PLC sublayer entity to reset itself exactly as at power-on. All LSAPs, except management LSAPs, are deactivated and all management LSAPs are activated by this request.

10B.3.2 L_STATUS.request

10B.3.2.1 Function. This primitive is the service request primitive for the L_status service.

10B.3.2.2 Semantics of the service primitive. The primitive shall provide parameters as follows:

L_STATUS.request
  (SSAP,
  DSAP,
  remote_address
  status_type)

SSAP specifies the local LSAP associated with the status request.

DSAP and remote_address specify the remote LSAP for which status is requested. If the status request is local, these parameters are null.

The status_type specifies the type of status being requested.

10B.3.2.3 When generated. This primitive is passed from the station management entity to the PLC entity to request local or remote status of an SAP component.

10B.3.2.4 Effect of receipt. Receipt of this primitive specifying local status will cause the local PLC entity to indicate the requested local status. Actions of the local PLC entity on receiving a request for remote status are for future study.

10B.3.3 L_STATUS.indication

10B.3.3.1 Function. This primitive is the service indication primitive for L_STATUS service.

10B.3.3.2 Semantics of the service primitive. The primitive shall provide parameters as follows:

L_STATUS.indication
(SSAP,
DSAP,
remote_address,
status_type)

SSAP specifies the local LSAP associated with the status indication.

DSAP and remote_address specify the remote LSAP for which status is requested. If the status request is local or the primitive is initiated by local spontaneous action, these parameters are null.

Status_type conveys the local or remote status depending upon whether the primitive is initiated by local spontaneous action, or by receipt of a status update from a remote LSAP.

10B.3.3.3 When generated. This primitive is passed from the PLC entity to the station management entity to inform the station management entity of the receipt of a remote status update, or to inform the station management entity of spontaneous changes within the PLC sublayer which require interaction with the upper layers and/or the station management entity.

10B.3.3.4 Effect of receipt. The effect of receipt of this primitive by the station management entity is for future study.

10B.3.4 L_STATUS.confirm

10B.3.4.1 Function. This primitive is the service confirmation primitive for the L_STATUS service.

10B.3.4.2 Semantics. The primitive shall provide parameters as follows:

L_STATUS.confirmation
(SSAP,
DSAP,
remote_address,
status_type,
status_value)

SSAP specifies the local LSAP associated with the status confirmation.

DSAP and remote_address specify the remote LSAP for which status was requested. If the status request is local, these parameters are null.

Status_type specifies the type of status being returned.

Status_value conveys the requested status information.

10B.3.4.3 When generated. This primitive is passed from the PLC entity to the station management entity to convey local or remote SAP status in response to an earlier request for status.

10B.3.4.4 Effect of receipt. The effect of receipt of this primitive by the station management entity is unspecified.

10B.3.5 L_SAP_ACTIVATE.request

10B.3.5.1 Function. This primitive is the service request primitive of LSAP activation and configuration service. This primitive deals with all components of the LSAP except those that respond to an RDR request.

10B.3.5.2 Semantics. The primitive shall provide parameters as follows:

L_SAP_ACTIVATE.request
  (SSAP,
  Services activated,
  Role in each service activated,
  Maximum L_sdu_length in each service activated)

SSAP specifies the local LSAP which is to be activated and configured.

Services activated specifies the services be supported by the SAP being activated. Allowed services are:

A given SAP may be configured to support one or more of these services in any combination.
SDA
SDN
RDR
non-PROWAY

Role specifies separately the role of this LSAP for each activated service. Allowed roles are:
INITIATOR
RESPONDER
BOTH = INITIATOR + RESPONDER

NOTE: That the RESPONDER role for the RDR service is configured with the L_RSAP_ACTIVATE.request.

Maximum L_sdu_length specifies separately for each activated service the maximum size of the L_data_unit exchanged across the PLC-user interface (Part 2A). The range of maximum L_sdu_length is 1 to 1000 octets.

10B.3.5.3 When generated. This primitive is passed from the station management entity to the PLC entity to activate and configure an LSAP, excluding its RDR response component.

10B.3.5.4 Effect of receipt. The receipt of this primitive by the PLC entity will cause activation of an LSAP with the specified LSAP address and configuration.

10B.3.5.5 Additional comments. The details of the "activation" are implementation-dependent.

Care should be taken that all SAPs specified in group DSAPs have been activated for the SDN service.

For efficiency, and for a well-structured system, a user should not be required to process the indication primitives from communication services it does not support. Hence the L_SAP_ Activate.request specifies the type(s) of service that a user entity expects to receive from PLC. This implies that if the L_SAP_Activate.request specifies only SDA/RESPONDER and a later

SDN L_pdu addressed to that LSAP is received by this (remote) PLC entity, the remote PLC entity will return an L_pdu with L_status = RS to the originating station. The remote PLC will not inform the remote user of the arrival of the SDN L_pdu.

NOTE: Management LSAPs must always be able to receive SDN Management_pdus. These management LSAPs are made active when the PLC entity is activated.

In Reference 3, Appendix I-A, an additional parameter may be supplied with the L_SAP_Activate.request to support a Link level retry procedure which is not used in PROWAY. For compatibility a PROWAY station may specify:

maximum_number_of_transmissions = 1

The L_SAP_Activate.request and the L_RSAP_Activate.request for an SAP must be associated to allow coordinated exits from the configure state as described in Part 10A and Appendix 10-B.

10B.3.6 L_SAP_ACTIVATE.confirm

10B.3.6.1 Function. This primitive is the service confirmation primitive for the SAP activation service.

10B.3.6.2 Semantics. The primitive shall provide parameters as follows:

L_SAP_Activate.confirm
(SSAP,
results)

SSAP indicates the local SAP for which the results are being conveyed.

The results parameter conveys the results of the previous associated L_SAP_Activate.request.

10B.3.6.3 When generated. This primitive is passed from the PLC entity to the station management entity to convey the results of the previous associated L_SAP_Activate.request primitive. The results indicate either that the SAP activation attempt was successful or that an activation of one or more of the services requested could not be achieved. In case of failure, results indicates those services whose activation failed.

10B.3.6.4 Effect of receipt. If the activation request was issued on behalf of a user, the receipt of this primitive by the station management causes it to issue an L_MGMT.confirm notifying the user of the success or failure of the activation.

10B.3.7 L_SAP_DEACTIVATE.request

10B.3.7.1 Function. This primitive is the service request primitive for the LSAP deactivation service.

10B.3.7.2 Semantics of the service primitive. The primitive shall provide parameters as follows:

L_SAP_Deactivate.request

(SAP)

The SAP parameter specifies the LSAP address which is to be deactivated.

10B.3.7.3 When generated. This primitive is passed from the station management entity to the PLC entity to deactivate an LSAP.

10B.3.7.4 Effect of receipt. The receipt of this primitive by the PLC sublayer will cause the deactivation of an SAP with the given LSAP address. It may optionally cause the purging of all outstanding service requests for this LSAP.

10B.3.7.5 Additional comments. The details of "deactivation" are implementation-dependent.

Deactivation will not occur while this LSAP has a request_with_response frame outstanding.

10B.3.8 L_SAP_DEACTIVATE.confirm

10B.3.8.1 Function. This primitive is the service confirmation primitive for the SAP deactivation service.

10B.3.8.2 Semantics. The primitive shall provide parameters as follows:

L_SAP_Deactivate.confirm
  (SAP
  results)

SAP indicates the local SAP for which the results are being conveyed.

The results parameter conveys the results of the previous associated L_SAP_Deactivate.request.

10B.3.8.3 When generated. This primitive is passed from the PLC entity to the station management entity to convey the results of the previous associated L_SAP_Deactivate.request primitive. The results indicate either that the LSAP deactivation attempt was successful or that a deactivation of one or more of the services requested could not be achieved. In case of failure, results indicate those services whose deactivation failed.

10B.3.8.4 Effect of receipt. If the deactivation request was issued on behalf of a user, the receipt of this primitive by the station management causes it to issue an L_MGMT.confirm notifying the user of the success or failure of the activation.

10B.3.9 L_RSAP_ACTIVATE.request

10B.3.9.1 Function. This primitive is the service request primitive for the RDR response component activation and configuration service.

10B.3.9.2 Semantics. The primitive shall provide parameters as follows:

L_SAP_Activate.request
  (SSAP,
  DSAP,
  shared_buffer_identification)

SSAP specifies the user which is to receive L_REPLY.indications when an RDR request is received at this DSAP.

DSAP specifies the DSAP number associated with the shared_buffer_area.

Shared_buffer_identification indicates the shared buffer which is made available for the RDR service. Its format is implementation-dependent.

10B.3.9.3 When generated. This primitive is passed from the station management entity to the PLC entity to activate and configure the RDR response component of an LSAP.

10B.3.9.4 Effect of receipt. The receipt of this primitive by the PLC entity will initiate the activation and configuration of the indicated RDR response component.

10B.3.9.5 Additional comments. The details of the "activation" are implement-dependent. The L_SAP_Activate.request and the L_RSAP_Activate.request for an SAP must be associated to allow coordinated exit from into the configure state as described in Part 10A and Appendix 10-B.

10B.3.10 L_RSAP_ACTIVATE.confirm

10B.3.10.1 Function. This primitive is the service confirmation primitive for the RDR response component activation and configuration service.

10B.3.10.2 Semantics. The primitive shall provide parameters as follows:

L_RSAP_Activate.confirm
  (SSAP,
  DSAP,
  results)

SSAP indicates the user which receives the L_REPLY.indication for when an RDR request is received at this DSAP.

DSAP indicates the DSAP for which the results are being conveyed.

The results parameter conveys the results of the previous associated L_RSAP_Activate.request.

10B.3.10.3 When generated. This primitive is passed from the station entity to the station management entity to convey the results of the previous associated L_RSAP_Activate.request primitive. The results indicate either that the DSAP configuration attempt was successful or that the configuration of the SAP could not be achieved.

10B.3.10.4 Effect of receipt. If the activation request was issued on behalf of a user, the receipt of this primitive by the station management entity causes it to issue an L_MGMT.confirm notifying the user of the success or failure of the activation.

10C. Station Management-MAC interface service specification

10C.1 Scope and field of application. This part specifies the services provided to the station management entity by the MAC sublayer of this standard. This standard specifies these services in an abstract way. It does not specify or constrain the implementation entities and interfaces within a computer system. The relationship of this part to other parts of this standard and to LAN specifications is illustrated in Figure 10C-1.

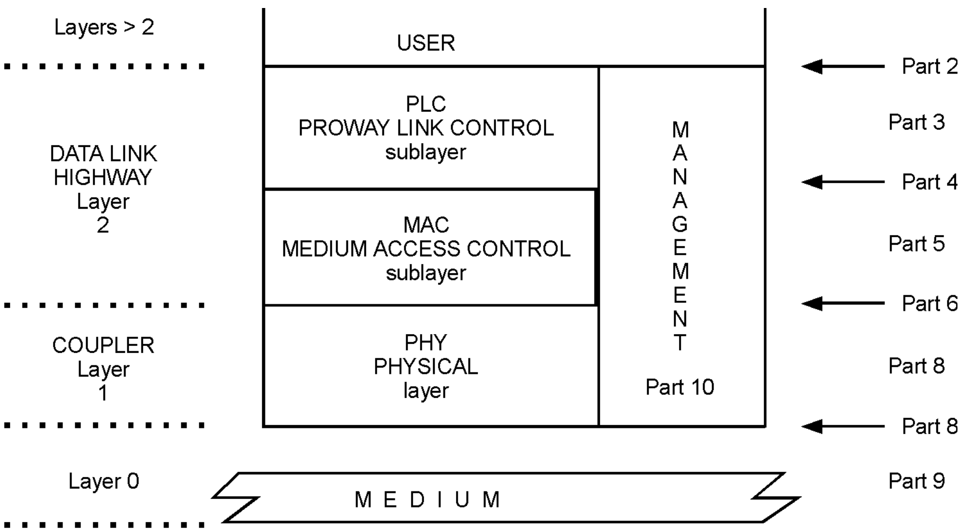


Figure 10C-1: Relationship to LAN model

10C.2 Overview of the station management-MAC service

10C.2.1 General description of services provided. This clause describes informally the services provided to and by the station management functions by and to the token-passing Medium Access Control sublayer of the Highway layer. These services are local administrative services between the MAC sublayer and its manager. They provide the means and method of

1) resetting the MAC entity, and selecting the MAC entity's MAC address (and implicitly the length of all MAC addresses on the network).

2) specifying the values of timer and counter preset constants appropriate for the network (for example, slot_time).

3) determining whether the MAC entity should be a member of the token-passing ring, and whether its address appears to be unique to the local network.

4) reading the current values of some of the MAC entity's parameters.

5) notifying the station management entity of relevant changes in the MAC entity's parameters.

6) specifying the set of group addresses which the MAC entity recognizes.

10C.2.2 Model used for the service specification. The model and descriptive method are detailed in Reference 1, Appendix I-A.

10C.2.3 Overview of interactions. The primitives associated with local administrative services (items 1-6 above) are:

MA_INITIALIZE_PROTOCOL.request
MA_INITIALIZE_PROTOCOL.confirmation

MA_SET_VALUE.request
MA_SET_VALUE.confirmation

MA_READ_VALUE.request
MA_READ_VALUE.confirmation

MA_EVENT.indication
MA_FAULT_REPORT.indication

MA_GROUP_ADDRESS.request
MA_GROUP_ADDRESS.confirmation

10C.2.3.1 MA_INITIALIZE_PROTOCOL. The MA_INITIALIZE_PROTOCOL.request primitive is passed to the MAC sublayer to reset the entire MAC sublayer and to select the MAC entity's MAC address (and implicitly the length of all MAC addresses on the network), the token-passing protocol appropriate for the network (that is, token_bus) and the station's role in that network (i.e., originate_only). It elicits an immediate MA_INITIALIZE_PROTOCOL.confirm specifying whether the desired protocol is available.

10C.2.3.2 MA_SET_VALUE. The MA_SET_VALUE.request primitive is passed to the MAC sublayer from the station management entity to set the value of a MAC variable. The MA_SET_VALUE.confirm returns a success or failure indication for the associated primitive.

10C.2.3.3 MA_READ_VALUE. The MAC_READ_VALUE.request primitive is passed to the MAC sublayer from the station management entity to request the value of a MAC variable. The MA_SET_VALUE.confirm returns the value of the requested MAC variable.

10C.2.3.4 MA_EVENT. The MA_EVENT.indication primitive is passed to the station management entity to indicate that the value of a significant MAC variable has changed.

10C.2.3.5 MA_FAULT_REPORT. The MA_FAULT REPORT.indication primitive is passed to the station management entity to indicate that an error has been inferred by the MAC entity.

10C.2.3.6 MA_GROUP_ADDRESS. The MA_GROUP_ADDRESS.request primitive is passed to the MAC sublayer to specify the set of MAC-level group addresses which the MAC entity should recognize, so that valid MAC-level frames with these destination addresses will be passed to the MAC user entity by way of the appropriate indicate primitive. The MA_GROUP_ADDRESS.confirm primitive returns a success or failure indication for the associated request primitive.

10C.2.4 Compliance. The functions of this section that apply to features implemented in this station are mandatory; however, the primitives and coding defined are for further study and harmonization with the IEEE 802 committee.

10C.3 Detailed interactions with the station management entity. This clause describes in detail the primitives and parameters associated with the MAC administrative services. Note that the parameters are specified in an abstract sense. The parameters specify the information that must be available to the receiving entity. A specific implementation is not constrained in the method of making this information available. For example, the station address (TS) or set-of-group-addresses parameters associated with some of the administrative service primitives maybe provided by actually passing MAC addresses, by passing descriptors, or by other means. The values of some selection parameters may be implied by an implementation. The MAC sublayer may also provide local confirmation mechanisms for all request type primitives.

10C.3.1 MA_INITIALIZE_PROTOCOL.request

10C.3.1.1 Function. This primitive is the service request primitive for the protocol initialization service. It also functions as a RESET service request primitive for the entire MAC sublayer.

10C.3.1.2 Semantics. This primitive shall provide parameters as follows:

MA_INITIALIZE_PROTOCOL.request
(desired_protocol)

Desired_protocol specifies the token-passing protocol which the MAC entity should implement; i.e., simple token_bus.

NOTE: That PROWAY stations always use the simple token_bus protocol.

10C.3.1.3 When generated. This primitive is passed from the station management entity to the MAC sublayer entity to request that the MAC sublayer entity reset itself and reconfigure itself as specified by the parameters.

10C.3.1.4 Effect of receipt. Receipt of this primitive causes the MAC entity to reset itself exactly as at power-on, select the desired protocol, and generate a MA_INITIALIZE_PROTOCOL.confirmation to indicate the availability of the desired protocol.

10C.3.1.5 Additional comments. This primitive causes the MAC entity to be a non-member of the token-passing ring.

10C.3.2 MA_INITIALIZE_PROTOCOL.confirmation

10C.3.2.1 Function. This primitive is the service confirmation primitive for the protocol initialization service.

10C.3.2.2 Semantics. This primitive shall provide parameters as follows:

MA_INITIALIZE_PROTOCOL.confirm
(status)

Status indicates the success or failure of the initialization request.

10C.3.2.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate the success or failure of the previous associated protocol initialization request.

10C.3.2.4 Effect of receipt. The effect of receipt of this primitive is unspecified.

10C.3.3 MA_SET_VALUE.request

10C.3.3.1 Function. This primitive is the service request primitive for the setting of values of MAC sublayer variables.

10C.3.3.2 Semantics. This primitive shall provide parameters as follows:

MA_SET_VALUE.request
(variable name,
desired_value)

Variable_name indicates which MAC sublayer variables are to be assigned the specified desired_ value. The following variables shall be settable by way of this primitive:

1) TS (station address, see 5C.1.1)
2) slot_time
3) hi_pri_token_hold_time
4) max_ac_4_rotation_time
5) max_ac_2_rotation_time
6) max_ac_0_rotation_time
7) max_ring_maintenance_rotation_time
8) ring_maintenance_initial_value
9) max_inter_solicit_count
10) in_ring_desired
11) event_enable_maxk (see 10C.3.7)
12) maximum_retry_limit

The ability to set other MAC variables is for future study.

The range of desired_value is the range defined in Part 5 for the specified MAC variable.

10C.3.3.3 When generated. This primitive is passed from the station management entity to the MAC sublayer entity to request that the MAC sublayer entity change the value of the specified variable.

10C.3.3.4 Effect of receipt. The receipt of this primitive causes the variable value to be changed and to generate the associated MA_SET_VALUE.confirm primitive.

10C.3.4 MA_SET_VALUE.confirm

10C.3.4.1 Function. This primitive is the service confirmation primitive for the MAC sublayer set value service.

10C.3.4.2 Semantics. This primitive shall provide parameters as follows:

MA_SET_VALUE.confirm
(status)

Status indicates the success or failure of the set value request. If the MAC variable of the MAC_SET_VALUE.request is not implemented, status shall be returned as failure.

10C.3.4.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate the success or failure of the previous associated MA_SET_VALUE.request.

10C.3.4.4 Effect of receipt. The effect of receipt of this primitive is unspecified.

10C.3.5 MA_READ_VALUE.request

10C.3.5.1 Function. This primitive is the service request primitive for the reading of values of MAC sublayer variables.

10C.3.5.2 Semantics. This primitive shall provide parameters as follows:

MA_READ_VALUE.request
(variable_name)

Variable_name indicates which of the MAC sublayer variables is to be read. Readable MAC sublayer variables are:

1) address of successor, NS (see 5D)
2) address of predecessor, PS (see 5D)
3) number of retries on request_with_response frames
4) in-ring.

Additional candidate variables under study are:

1) number of stations in ring
2) measured token rotation time
3) number of valid received frames
4) number of received frames with FCS errors.

The ability to read other variable names is for future study.

The range of values associated with each variable is given in Part 5.

10C.3.5.3 When generated. This primitive is passed from the station management entity to the MAC sublayer entity to request that the MAC sublayer entity return the value of the specified variable.

10C.3.5.4 Effect of receipt. The receipt of this primitive causes the value of the specified variable to be returned by way of the associated MA_READ_VALUE.confirm primitive.

10C.3.6 MA_READ_VALUE.confirm

10C.3.6.1 Function. This primitive is the service confirmation primitive for the read value service.

10C.3.6.2 Semantics. This primitive shall provide parameters as follows:

MA_READ_VALUE.confirm
(variable_name,

current_value,
status)

If status equals success, the MAC variable, variable_name, has taken on the current_value specified in the associated read value primitive. If status equals failure, the MAC variable is not readable or not implemented and the value of current_value is not specified.

10C.3.6.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate the success or failure of the previous associated MA_READ_VALUE.request.

10C.3.6.4 Effect of receipt. The effect of receipt of this primitive is unspecified.

10C.3.7 MA_EVENT.indication

10C.3.7.1 Function. This primitive is the service indication primitive by which station management is informed of significant events within the MAC sublayer.

10C.3.7.2 Semantics. This primitive shall provide parameters as follows:

MA_EVENT.indication
(event)

Event identifies the event that has occurred with the MAC sublayer. The events that cause an MA_EVENT.indication are:

1) change of successor address
2) change of successor address to null.

Additional events which lead to MA_EVENT.indications are for future study.

10C.3.7.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate the occurrence of an enabled significant event within the MAC sublayer. It is enabled when an MA_SET_VALUE.request has set the event_enable mask to a non-zero value.

10C.3.7.4 Effect of receipt. The effect of receipt of this primitive is on the station management entity and is specified in Part 10A.

10C.3.8 MA_FAULT_REPORT.indication

10C.3.8.1 Function. This primitive is the service indication primitive for MAC failure indications.

10C.3.8.2 Semantics. This primitive shall provide parameters as follows:

MA_FAULT_REPORT.indication
(fault_type)

Fault_type identifies the particular fault condition the MAC layer has detected. The events which cause an MA_FAULT_ REPORT.indication are:

1) duplicate_address
2) faulty_transmitter

The duplicate_address fault is indicated when the MAC sublayer entity has inferred that another MAC entity on the network which has the same MAC address as the current value of the variable TS.

The faulty_transmitter fault is indicated when the MAC sublayer entity has inferred evidence that the station's transmitter is not being received correctly by other stations in the network.

10C.3.8.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate that the MAC entity has detected a failure condition within the protocol. The indication is passed when the access control machine transitions to the offline state.

The situations in which an MA_FAULT_REPORT.indication is passed are defined in the ACM state tables in Part 5C.

10C.3.8.4 Effect of receipt. The effect of receipt of this primitive is on the station management entity and is specified in Part 10A.

10C.3.8.5 Additional comments. This primitive is generated in response to detection of a network administration or hardware fault which may be due to failure of circuitry in either the detecting station or another station.

10C.3.9 MA_GROUP_ADDRESS.request

10C.3.9.1 Function. This primitive is the service request primitive for the protocol's group address activation service.

10C.3.9.2 Semantics. This primitive shall provide parameters as follows:

MA_GROUP_ADDRESS.request
  (set of group-addresses)

Set of group-addresses specifies a group of zero or more MAC entity addresses.

10C.3.9.3 When generated. This primitive is passed from the station management entity to the MAC sublayer entity to request that the MAC sublayer entity recognize the specified set of group addresses, so that valid MAC-level frames with these destination addresses will be passed to LLC, PLC or station management by way of the appropriate indication primitive.

10C.3.9.4 Effect of receipt. Receipt of this primitive causes the MAC sublayer entity to load the desired set of group addresses for comparison with the destination address of LLC, PLC and station management frames and to return a MA_GROUP_ADDRESS.confirm primitive. The last set of group addresses loaded are the only ones recognized. Loading zero (no) group addresses will deactivate the group address recognition service.

10C.3.9.5 Additional comments. The predefined broadcast group address (all address bits = one) is always recognized and cannot be disabled.

10C.3.10 MA_GROUP_ADDRESS.confirm

10C.3.10.1 Function. This primitive is the confirmation primitive for the protocol's group address activation service.

10C.3.10.2 Semantics. This primitive shall provide parameters as follows:

MA_GROUP_ADDRESS.confirm
  (status)

Status indicates the success or failure of the request.

10C.3.10.3 When generated. This primitive is passed from the MAC sublayer entity to the station management entity to indicate the success or failure of the previous associated MA_GROUP_ADDRESS.request.

10C.3.10.4 Effect of receipt. The effect of receipt of this primitive is unspecified.

NOTE: To provide a basis for network monitoring and analysis, an implementation also may support the following:

1 Reception of all data_frames, independent of destination address
2 Reception of all frames, including MAC_control frames, as specified in 5D.1.3.

10D. Station Management-Physical layer interface service specification

10D.1 Scope and field of application. This section specifies the services provided to the station management entity by the Physical layer entity of this standard. The relationship of this section to other sections of this standard and to LAN specifications is illustrated in Figure 10D-1.

10D.2 Overview of the Physical layer-management service

10D.2.1 General description of services provided at interface. These paragraphs describe informally the services provided to the station management entity by the Physical layer entity. These services are all local administrative services between the Physical layer and its manager. This set of services provides the means and method of:

1) resetting the Physical layer entity

2) determining the available and current operating modes of the Physical layer entity and selecting the appropriate operating modes. Modal choices can include:

   a) Transmitter output disable/enable (per drop cable)

   b) Received signal source (either a drop cable or a specified loopback point)

   c) Received signal level reporting

3) notifying station management of changes in current operating modes not caused by a PHY_MODE.select request.

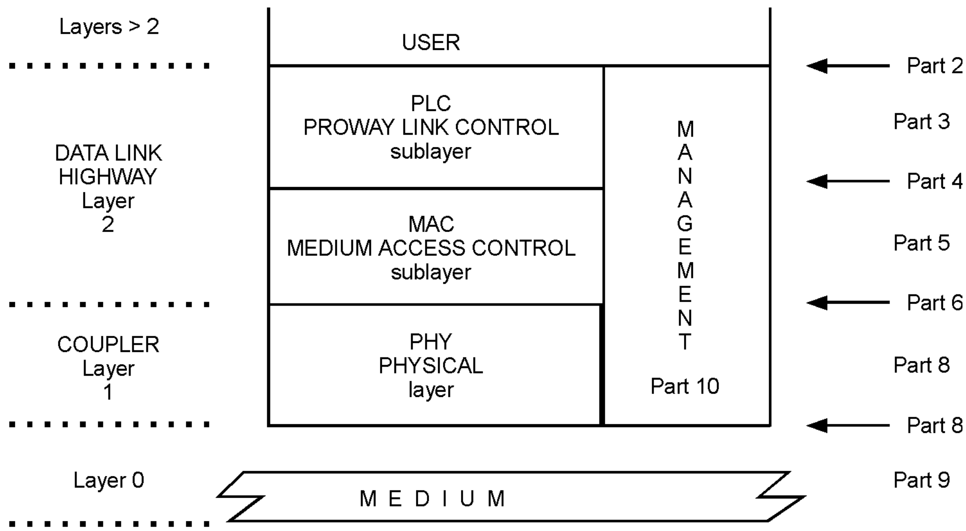


Figure 10D-1: Relationship to LAN model

10D.2.2 Model used for the service specification. The model and descriptive method are detailed in Reference 1, Appendix I-A.

10D.2.3 Overview of interactions. The primitives associated with local administrative services are

PHY_RESET.request
PHY_RESET.confirm
PHY_MODE_SELECT.request
PHY_MODE_SELECT.confirm
PHY_MODE_CHANGE.indication

10D.2.3.1 PHY_RESET request and confirm. The PHY_RESET.request primitive is passed to the Physical layer to reset the physical entity. It elicits an immediate PHY_RESET.confirm indicating success or failure of the reset.

10D.2.3.2 PHY_MODE_SELECT request and confirm. The PHY_MODE_SELECT.request primitive is passed to the Physical layer to select the Physical layer entity's mode of operation with respect to a specified class of modal operation. It elicits an immediate PHY_MODE_SELECT. confirm primitive specifying the success or failure of the selection request.

10D.2.3.3 PHY_MODE_CHANGE.indication. The PHY_MODE_CHANGE.indication primitive is passed to the station management entity to notify it of the occurrence of a non-commanded change in one of the Physical layer entity’s modes of operation (for example, transmitter disabling by way of activation of the Physical layer entity's jabber inhibit function).

10D.2.4 Compliance. The functions of this section that apply to features implemented in this station are mandatory; however, the primitives and coding defined are for further study and harmonization with the IEEE 802 committee.

10D.3 Detailed specifications of interactions with the station management entity.

This subsection describes in detail the primitives and parameters associated with the Physical layer administrative services. Note that the parameters are specified in an abstract sense. The parameters specify the information that must be available to the station management entity. A specific implementation is not constrained in the method of making this information available. For example, the representation of received signal source is not specified.

10D.3.1 PHY_RESET.request

10D.3.1.1 Function. This primitive is the service request primitive for the Physical layer reset service.

10D.3.1.2 Semantics. This primitive shall be parameterless, as follows:

PHY_RESET.request

10D.3.1.3 When generated. This primitive is passed from the station management entity to the Physical layer entity to request that the Physical layer entity reset itself.

10D.3.1.4 Effect of receipt. Receipt of this primitive causes the Physical layer entity to reset itself exactly as at power-on.

10D.3.2 PHY_RESET.confirm

10D.3.2.1 Function. This primitive is the service confirmation primitive for the Physical layer reset service.

10D.3.2.2 Semantics. This primitive shall provide parameters as follows:

PHY_RESET.confirm
  (LAN_topology_type,
  PHY_role)

LAN_topology_type specifies that the associated LAN is a token_bus (on a broadcast medium).

PHY_role specifies the Physical layer entity should function as an originate_only station.

10D.3.2.3 When generated. This primitive is passed from the Physical layer entity to the station management entity in confirmation to the previous associated physical reset request.

10D.3.2.4 Effect of receipt. The station management entity uses the LAN topology type and PHY role information to determine which protocol the associated MAC entity should employ.

10D.3.3 PHY_MODE_SELECT.request

10D.3.3.1 Function. This primitive is the service request primitive for the Physical layer current_ mode select service.

10D.3.3.2 Semantics. This primitive shall provide parameters as follows:

PHY_MODE_SELECT.request
(mode_class,
new_mode)

mode_class may specify one of the classes listed below. Other values of the mode_class parameter will elicit a failure confirmation status.

1) transmitter_outputs_inhibit

2) received_signal_source

new_mode specifies the desired mode of the designated mode_class. Allowed values for each mode_class:

1) transmitter_output_inhibits:

True — transmitter is inhibited from outputting to its associated medium
False — transmitter is enabled to output to its associated medium

NOTE: In a PHY entity with multiple transmitter this primitive indicates the desired state of each transmitter.

2) received_signal_source:

Primary —	station is to listen to the primary medium (primary receiver = ON, alternate = OFF)
Alternate —	station is to listen to the alternate medium (primary receiver = OFF, alternate receiver = ON)
Looped —	station is to looping back from within PHY layer (both receivers = OFF)

NOTE: In a PHY entity with multiple loopback points this primitive indicates the loopback point desired.

10D.3.3.3 When generated. This primitive is passed from the station management entity to the Physical layer entity to request that the Physical layer entity change its current operating mode of the designated mode_class to the mode specified by new_mode.

10D.3.3.4 Effect of receipt. Receipt of this primitive causes the Physical layer entity to attempt to change its operating mode of the designated mode_class to the specified mode, and to generate a PHY_MODE_SELECT.confirm primitive to indicate the status of the requested change.

NOTE: When a looped back condition exists in the PHY entity, all transmitters must be inhibited.

10D.3.4 PHY_MODE_SELECT.confirm

10D.3.4.1 Function. This primitive is the service confirmation primitive for the Physical layer current_mode select service.

10D.3.4.2 Semantics. This primitive shall provide parameters as follows:

PHY_MODE_SELECT.confirm
(mode_class,
status)

mode_class may take any of the values specified in 10D.3.3. All other values of the mode_class parameter shall elicit a **failure** confirmation status.

Status indicates the **success** or **failure** of the operating mode selection request.

10D.3.4.3 When generated. This primitive is passed from the Physical layer entity to the station management entity in confirmation to the previous associated mode select request to indicate the success or failure of that request.

10D.3.4.4 Effect of receipt. The effect of receipt of this primitive by the station management entity is unspecified.

10D.3.5 PHY_MODE_CHANGE.indication

10D.3.5.1 Function. This primitive is the service indication primitive for the Physical layer noncommanded mode change notification service.

10D.3.5.2 Semantics. This primitive shall be parameterless, as follows:

PHY_MODE_CHANGE.indication

10D.3.5.3 When generated. This primitive is passed from the Physical layer entity to the station management entity to indicate that one or more of the Physical layer entity's current modes has changed since the previous PHY_MODE_SELECT.confirm or PHY_MODE_CHANGE.indication primitive.

10D.3.5.4 Effect of receipt. The effect of receipt of this primitive by the station management entity is unspecified.

10D.3.5.5 Additional comments. This primitive is used to provide timely notification of noncommanded mode changes, such as automatic transmitter_output_inhibit due to the activation of the Physical layer's jabber inhibit function.

10E. Token ring management

10E.1 Scope and field of application. This part specifies those activities of the management entity of a station that participates in the token ring that relate to maintenance and reporting of the active_station_list. These activities are referred to as Token Ring Management.

This standard specifies these activities informally and in an abstract way. It does not specify or constrain the implementation of entities and interfaces within a computer system. The relationship of this section to other sections of this standard and to LAN specifications is illustrated in Figure 10E-1.

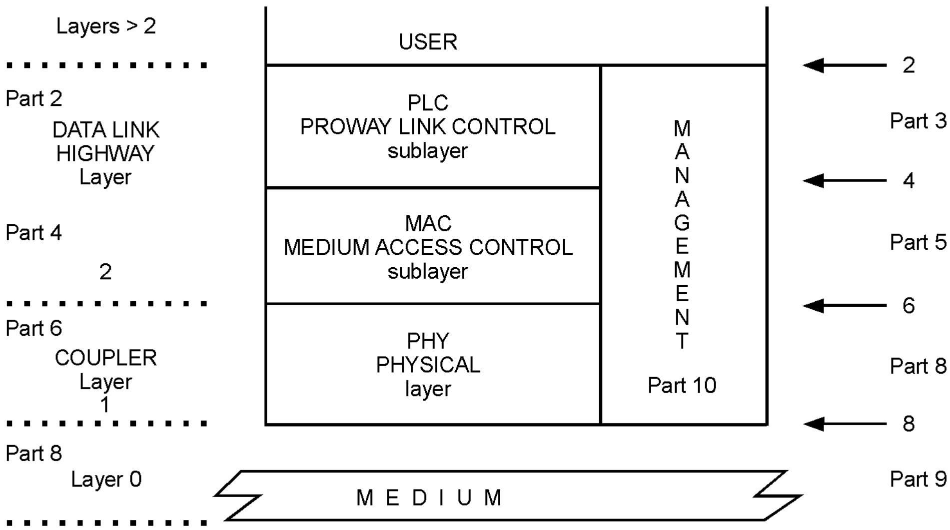


Figure 10E-1: Relationship to LAN model

10E.2 Overview of token ring management. The management entity of a PROWAY station that participates in the logical token ring has responsibilities related to ring participation. These are:

1) managing the station's participation in the logical token ring

2) maintenance of the active_station_list

3) reporting membership and changes in the active_station_list to PROWAY users.

10E.2.1 Management of ring participation. The station's management of participation in the logical token ring is described in Part 10A. Recommendations to supplement Part 10A are informally given in Appendix 10B.

The PROWAY user may request management transitions to the configure, line_connect and the line_disconnect states by primitives passed across the user-management interface described in Part 2B. The configure state is a stable, non-communicating state of an LSAP component of the PLC entity which allows setting of parameters which affect the station's interaction with other participating stations. The line_connect state provides a means for an initialization of the PLC, MAC and PHY entities and for entry into the logical token ring. The line_disconnect state provides a means for orderly exit from the token ring.

10E.2.2 Maintenance of the active_station_list. The management entity of each station that participates in the token ring maintains the list of stations participating in that logical token ring. This list is known as the active_station_list. A station which is currently cooperating in the token-passing process, i.e., one for which the MAC variable **in-ring** is true, is known as an active station and should be included in the active_station_list. Each station's management entity maintains an active_station_list. It does this in cooperation with the management entities of all other stations participating in the token ring by receiving, adjusting, recording, and passing on the current active_ station_list and active_station_change notifications. The methods for creating and maintaining this list, and the responsibilities of each station's management entity, are specified in 10E.4.

10E.2.3 Report of active_station_list membership. A user may submit an L_MGMT.request for a report of all stations currently participating in the token-passing process as described in Part 2B. The list will be returned by an L_MGMT.confirm as described in Part 2B.

10E.2.4 Report of spontaneous changes in active_station_list membership. A particular designated user identified by SSAP = 01100000 will be notified by an L_MGMT.indication of all changes in active_station_list membership as described in Part 2B. The management activities in generating the L_MGMT.indication are described in 10E.4.

10E.3 Compliance. Implementation of the protocol and frame formats of Part 10E are strongly recommended for all PROWAY stations that participate in the logical token ring, i.e., all **INITIATORS**.

NOTE: Proper operation of this protocol requires participation by all token-holding stations.

10E.4 Maintenance of the active_station_list

10E.4.1 Informal overview of active_station_list maintenance procedures

Change Notifications. When a station detects a change in its NS, it notifies its new NS of the change. This initiates a series of change notifications that travel in turn to each station in the token ring. At each station the notification is used to update that station's active_station_list. The notification is also passed to that station's NS if required. Change notifications for more than one change may be circulating around the token ring simultaneously.

Creation of active_station list. The active_station_list is initially created by a bootstrap procedure involving the change notifications.

Maintenance of active_station_list. The active_station_list is maintained by each station as it receives change notifications.

Receiving an Initial active_station_list. Stations which have never been in the ring or which have left the ring have no active _ station_list except TS. A station entering (or re-entering) the ring receives an initial active_station_list from its PS.

Removal of change notifications. When a station receives a change notification which is redundant (i.e., its list is already correct with respect to the change) it does not continue to circulate the change. This insures that the change circulates to each station only once. The redundant notification will usually be removed by its initiator but may be removed by a successor if the initiator leaves the ring before the notification makes one round of the ring.

Transmission of active_station_lists and change notifications. Request_with_response procedures are used to prevent aliasing of the time order of change notifications. Priority = 4 is used to allow timely notification of changes.

Interactions with participation in the logical token ring. Recommended interactions with entry into and exiting from the token ring are given in Appendix 10-B.

10E.4.2 Detailed active_station_list maintenance procedures. The management entity of each station that participates in the logical token ring must cooperate in maintaining the active_station_list. The following protocol must be followed by each individual station's management entity:

1) When the management entity enters the line_connect state it must initialize its active_station_list to TS only.

2) When the management entity receives an MA_EVENT.indication indicating that NS has changed:

IF event = NS_changed_to_null [i.e., to out-of-ring] THEN initialize active_station_list to TS only

ELSE [i.e., NS changed to non-null address]

IF current active_station_list ≠TS [i.e., now in-ring]

IF (new_NS > old_NS) OR
(new_NS < any station_address in current active_station_list) [i.e., new station added to ring]

a) Add address of new_NS to current active_station_list.

b) Send complete active_station_list to new_NS.

c) Generate an L_MGMT.indication of the change in the active_station_list.

ELSE [i.e., station or stations removed from ring]

a) Send active_station_delete notification to new NS. This notification includes the addresses of all stations in current active_station_list for which TS > station_address > new_NS. No notification is sent if no stations meet this new criteria.

b) Remove all stations from active_station_list for which TS > station_address > new_NS

c) Generate an L_MGMT.indication of the change in the active_station_list.

END IF

ELSE [i.e., not now in-ring]

a) Start an active_station_list_timer which will expire in approximately 1 second.

END IF

END IF

3) When an active_station_add_notification is received from PS FOR each address contained in the active_station_add_notification,

IF this station is NOT in the current active_station_list,

a) Add station to active_station_list.

b) Send an active_station_add_notification for this station to NS.

c) Generate an L_MGMT.indication of the change in the active_station_list.

ELSE

(The notification is redundant and no action is taken.)

END IF

NEXT station_address

4) When an active_station_delete_notification is received from PS FOR each address contained in the active_ station_ delete_ notification,

IF this station is in the active_station_list,

a) Delete station from active_station_list.

b) Send an active_station_delete_notification for this station to NS.

c) Generate an L_MGMT.indication of the change in the active_station_list.

ELSE

(The notification is redundant and no action is taken.)

END IF

NEXT station_address

5) When an active_station_list is received from PS,

a) Store the received list as the current active_station_list.

b) Send an active_station_add_notification for TS to NS.

c) Stop the active_station_list_timer.

6) When the active_station_list_timer expires [i.e., PS left the ring before transmitting an initial active_station_list],

a) Determine address of PS using an MA_READ__VALUE.request and confirmation.

b) Send an active_station_list_request to PS.

c) Restart the active_station_list_timer.

10E.4.3 Communications with the management entities of other stations

10E.4.3.1 Transmission of active_station_list and change_notifications. The active_station_list and active_station_change_notifications are sent to this station's NS using the SDA service at Priority 4. The SSAP and DSAP for these transmissions shall equal 01110000.

10E.4.3.1.1 Format of transmitted active_station_list. For the active_station_list, the first octet of the Mgt_pdu shall have a value of **0**. The remainder of the Mgt_pdu is a list of the station addresses of all stations in the current active_station_list of the initiating management entity. Station addresses shall be arranged in ascending numerical order to facilitate storage and examination of members of the list. Station addresses shall be in the format of Part 5D.

10E.4.3.1.2 Format of transmitted active_station_add_notification. For active_station_add notifications, the first octet of the Mgt_pdu shall have a value of **1**. The remainder of the Mgt_pdu shall contain station addresses of stations to be added to the active_station_list. Station addresses shall be arranged in ascending numerical order. Station addresses shall be in the format of Part 5D and Appendix 5-B.

10E.4.3.1.3 Format of transmitted active_station_delete_notification. For active_station_delete_notifications, the first octet of the Mgt_pdu shall have a value of **2**. The remainder of the Mgt_pdu shall contain station addresses of stations to be deleted from the active_station_list. Station addresses shall be arranged in ascending numerical order. Station addresses shall be in the format of Part 5D.

10E.4.3.2 Transmission of active_station_list_requests. Active_station_list_requests are sent to this station's PS using the SDN service at Priority 4. The SSAP and DSAP for these transmissions shall equal 01110000. The Mgt_pdu shall consist of one octet which shall have a value of **3**.

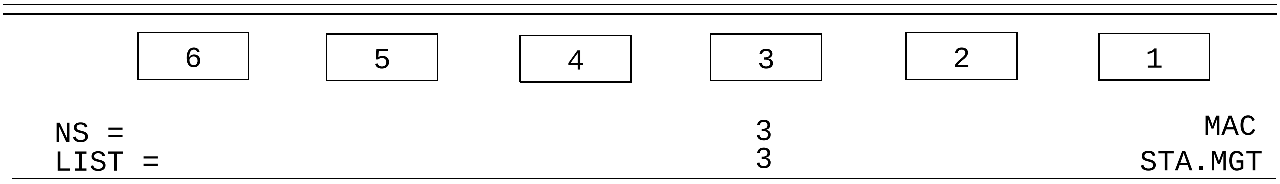
10F. Reset management

The subject of resetting a station's communication function is for future study.

Appendix 10-A Examples of active station list maintenance

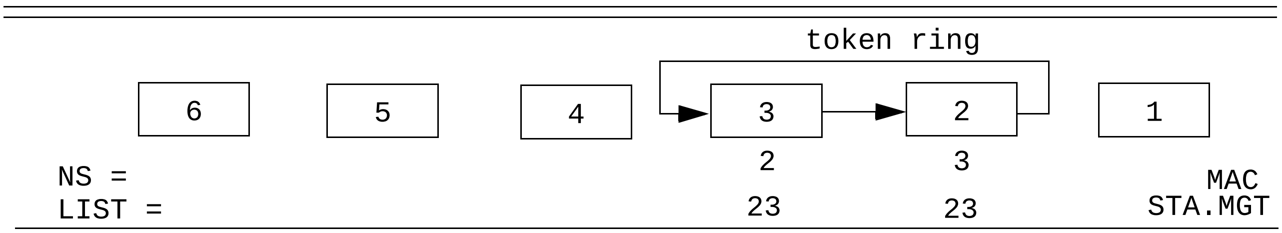
This appendix gives examples of the operation of active_ station_ list maintenance procedures as specified in Part 10E. This appendix is **not** mandatory.

Starting Condition-Station 3 only = on line



Event-Station 2 comes on line

Resulting Condition



Description:

a) Station 2 comes on line and is added to token ring

b) Station 3 receives MA_EVENT.indication with new_NS = 2

c) Station 3 adds 2 to active_station_list

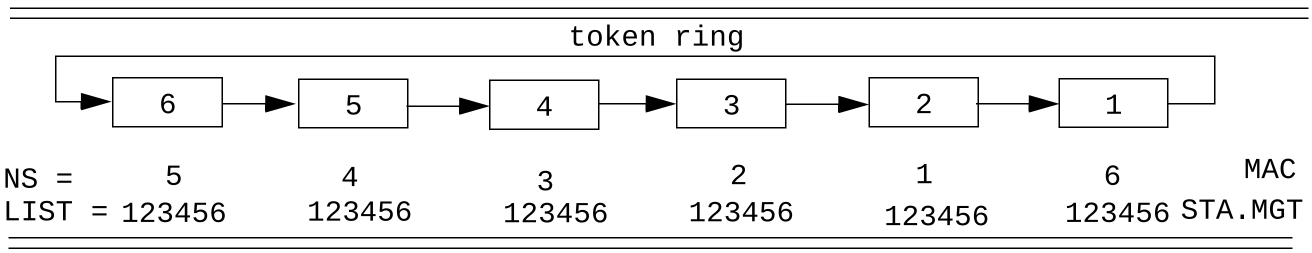
d) Station 3 sends active_station_list to its new_ NS (Station 2)

e) Station 2 receives and saves the active_station_list

f) Station 2 sends notification for station 2 to its NS (Station 3)

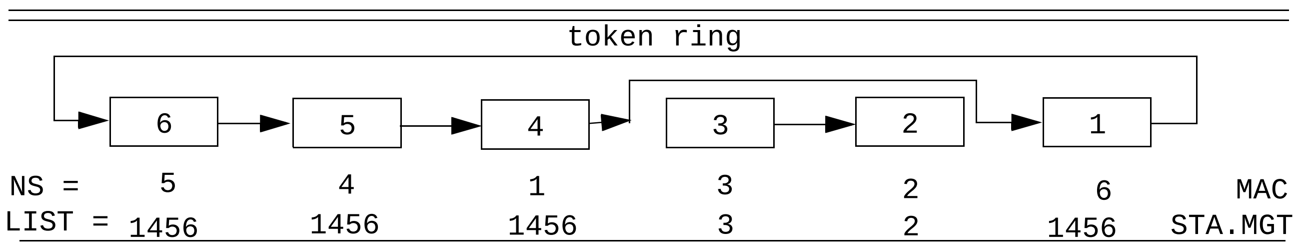
g) Station 3 receives the notification and detects that it is redundant (i.e., Station 2 is already in the list) and so ends the cycle and the active_station_list is formed.

Starting Condition-All Stations Active



Event-Stations 2 and 3 leave the ring

Resulting Condition



Description:

a) Stations 2 and 3 leave the ring

b) Station 4 management entity receives MA_EVENT.indication with new_NS = 1. This condition is new_NS < old_NS indicating a station or stations were deleted from the ring

c) Station 4 deletes from its active_station_list all station addresses such that TS > station address > new_NS (i.e., 2 and 3)

d) Station 4 sends an active_station_delete_notification to its new_NS listing all stations from its active_station_list such that TS > address > new_NS (i.e., 2 and 3)

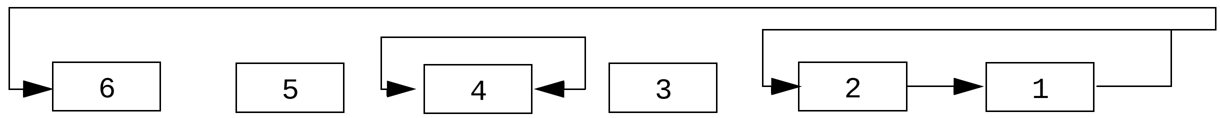
e) Station 1 receives the active_station_delete_notification from 4 and deletes the listed stations from its active_station_list

f) Station 6 receives the active_station_delete_notification from 1 and deletes the listed stations from its active_station_list

g) Station 5 receives the active_station_delete_notification from 6 and deletes the listed stations from its active_station_list

h) Station 4 receives the active_station_delete_notification from 5, detects that the listed deletions are redundant, and ends the cycle by ignoring the notification

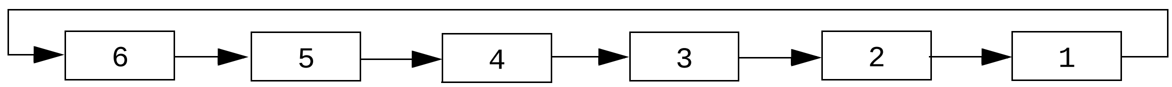
Starting Condition-Stations, 1, 2, 4, 6 Active



NS =	4	5	2	3	1	6	MAC
LIST =	1246	5	1246	3	1246	1246	STA.MGT

Event-Stations 3 and 5 enter the ring simultaneously

Resulting Condition-All Stations Active



NS =	5	4	3	2	1	6	MAC
LIST =	123456	123456	123456	123456	123456	123456	STA.MGT

Description:

Notes: (a) is the common activity of both changes occurring, simultaneously (b) and all items at left margin are results of 5 entering the ring (c) and all indented items are the result of 3 entering the ring

a) Stations 3 and 5 enter the ring at essentially the same instant

b) Station 6 receives a MA_EVENT.indication with new_NS = 5 and begins a recycle by sending its active_station_list to station 5

c) Station 4 receives MA_EVENT.indication with = 3 and begins a cycle by sending its active_station_list to station 3

d) Station 5 receives the list from 6 and saves it

e) Station 5 sends active_station_add_notification to station 4 to add 5

f) Station 3 receives the active_station_list from 4 and saves it

g) Station 3 sends an active_station_add_notification to station 2 to add 3

h) Station 4 adds 5 to its active_station_list and passes on the add 5 notification to 3

i) Station 2 adds 3 to its active_station_list and passes on the add 3 notification to 1

j) Station 3 adds 5 to active_station_list and passes on the add 5 notification to 2

k) Station 1 adds 3 to active_station_list and passes on the add 3 notification to 6

l) Station 2 adds 5 to active_station_list and passes on the add 5 notification to 1

m) Station 6 adds 3 to active_station_list and passes on the add 3 notification to 5

n) Station 1 adds 5 to active_station_list and passes on the add 5 notification to 6

o) Station 5 adds 3 to active_station_list and passes on the add 3 notification to 4

p) Station 6 receives the add 5 notification and stops the cycle because the notification is redundant

q) Station 4 receives the add 3 notification and stops the cycle because the notification is redundant

Appendix 10-B Recommendations for management of ring participation and redundant media

This appendix gives recommendations for management of a station's participation in the token ring and for management of redundant media. These recommendations are **not** mandatory.

10-B.1 Recommendations for ring participation management

10-B.1.1 Management of ring participation. The station's management of participation in the logical token ring is described in Part 10A. Recommendations to supplement Part 10A are informally given in Appendix 10-B.

The PROWAY user may request management transitions to the configure, line_connect and the line_disconnect states by primitives passed across the user-management interface described in Part 2B. The configure state is a stable, non-communicating state of an LSAP component of the PLC entity which allows setting of parameters which affect the station's interaction with other participating stations. The line_connect state provides a means for an initialization of the PLC, MAC and PHY entities and for entry into the logical token ring. The line_disconnect state provides a means for orderly exit from the token ring.

10-B.1.2 Ring participation management station descriptions. This section provides an informal description of the state diagram of Figure 10-2.

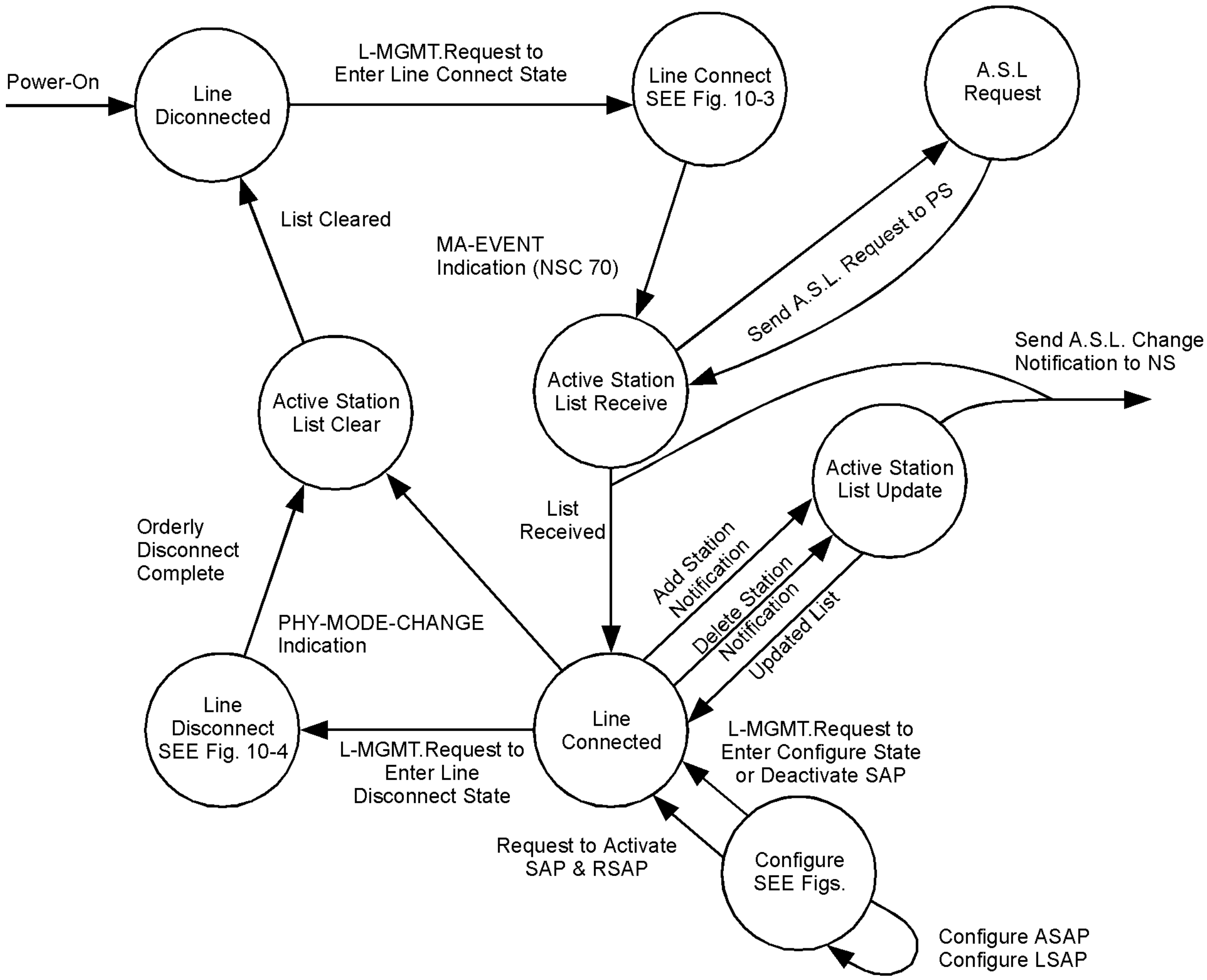


Figure 10-2: Ring participation management

10-B.1.3 Line_disconnect state. The line_disconnect state is entered after the completion of an orderly disconnect in response to a user L_MGMT.request to enter line_disconnect state. The transitions required to achieve an orderly disconnection are described in Figure 10-3.

The line_disconnect state exits to the line_connect state in response to a user L_MGMT.request to enter line_connect state as described in Part 10A.

10-B.1.4 Line_connect state. The line_connect state is entered from the line_disconnect state in response to a user L_MGMT.request to enter the line_connect state. This is the user's method of requesting connection to the network. Entry to the line_connect state starts the process of establishing an orderly connection to the network. Prior to attempting a connection, the PLC, MAC and PHY entities are initialized as described in Part 10A.4.2. The transitions required to achieve an orderly connection are described in Figure 10-4.

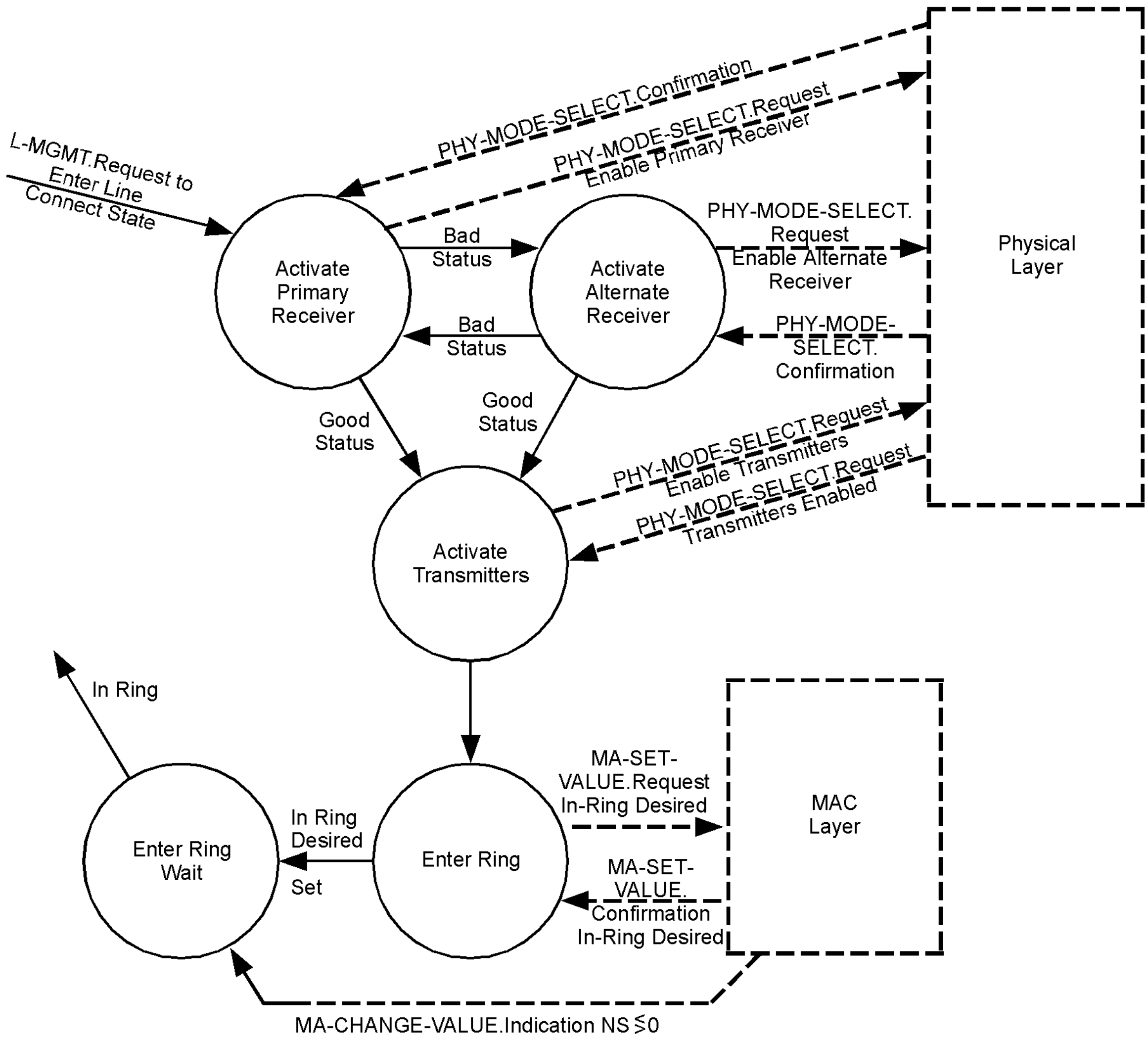


Figure 10-3: Line_connect state diagram

The line_connect state exits to the active_station_list_receive state after the orderly connection to the network is complete. The details of this condition are described in Figure 10-4.

10-B.1.5 Active_station_list_receive state. The active_station_list_receive state is entered from the line_connect state after the orderly connection of the station as detailed in Figure 10-4. The station waits in this state for an active_station_list to be received from the station's PS as described in Part 10E.

The active_station_list_receive substate is exited to the line_connect state upon receipt of an active_station_list.

10-B.1.6 Line_connect state. The line_connect state is entered from the active_station_receive state upon receipt of an active_station_ list from the station's PS. This is the nominal state allowing communication by this station on the local area network

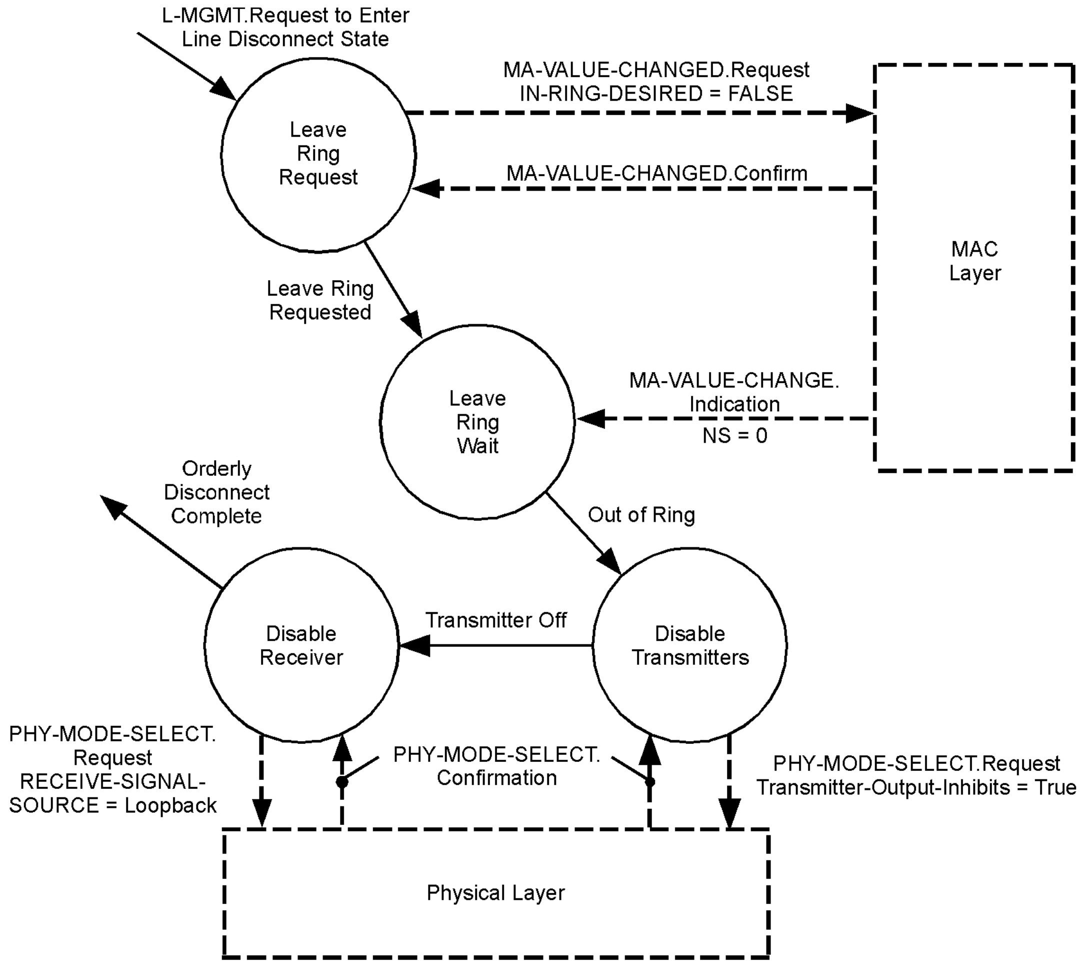


Figure 10-4: Line_disconnect state diagram

The line_connect state may exit to one of four states as described below.

1) Receipt of an active_station_add_notification or active_station_delete_notification will cause an exit from the line_connect state to the active_station_list update substate.

2) Receipt of a PHY_MODE_CHANGE.indication will cause an exit from the line_connect state to the active_station_list_clear state. This indication is received due to a "jabber halt" condition as described in Part 10D.

3) The user may cause an exit from the line_connect state by issuing an L_MGMT.request to enter the line_disconnect state or configure state.

4) The user may request deactivation of a PLC SAP component by issuing an L_SAP_Deactivate.request.

10-B.1.7 Active_station_list_update substate. The station enters the active_station_list_update substate from the line_connect state on receipt of an active_station_add_notification or active_station_delete_notification from the station's PS as described in Part 10E. Note that entry to or exit from the active_station_list_update substate does not affect operation of the PLC entity.

The station's copy of the active_station_list is updated while in this state according to the protocol described in Part 10E. The station then returns to the line_connect state.

10-B.1.8 Configure state. The configure state is entered for all supported specified PLC SAP components on receipt from the user of an L_MGMT.request to enter configure state. It is entered for the specified PLC SAP component on receipt of an L_SAP_Deactivate.request. These user requests are only allowed in the line_connect state. The enter_configure state command is used after initial entry to the line_connect state as described in 10-B.1.6 above. L_SAP_Deactivate.requests are used to disable or reconfigure a currently active SAP component.

PLC SAPs component may be activated while in the configure state relative to that SAP component. The configure state exits to the line_connect state for each SAP when that SAP has been activated by the L_MGMT.requests described in Parts 10A and 10B. The configure state of the station management entity corresponds to the configure state of the local and remote PLC state machines described in Part 3B.

10-B.1.9 Line_disconnect state. The line_disconnect state is entered in response to a user L_MGMT.request to enter line_disconnect state. This initiates an orderly disconnect as described in Figure 10-4. This includes a notification to PS that this station desires to leave the ring. Notification of PS is achieved by issuing an MA_VALUE_CHANGE.request for in_ring_desired equals **false**.

10-B.2 Recommendation for management of redundant media. Redundant media are best managed by transmitting on all available media while selecting one functional medium for reception. State transitions to accomplish this goal are outlined in Figures 10-3 and 10-4.

In addition, media not currently in use for reception should be tested and the user notified of any failures. These activities are for further study.

Developing and promulgating technically sound consensus standards, recommended practices, and technical reports is one of ISA's primary goals. To achieve this goal the Standards and Practices Department relies on the technical expertise and efforts of volunteer committee members, chairmen, and reviewers.

ISA is an American National Standards Institute (ANSI) accredited organization. ISA administers United States Technical Advisory Groups (USTAGs) and provides secretariat support for International Electrotechnical Commission (IEC) and International Organization for Standardization (ISO) committees that develop process measurement and control standards. To obtain additional information on the Society's standards program, please write:

ISA
Attn: Standards Department
67 Alexander Drive
P.O. Box 12277
Research Triangle Park, NC 27709

ISBN: 0-87664-896-0