



**IEEE Standard for
Information technology—
Telecommunications and information
exchange between systems—
Local and metropolitan area networks—
Specific requirements**

Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications

Amendment 3: Frame format extensions

IEEE Computer Society

Sponsored by the
LAN/MAN Standards Committee

IEEE
3 Park Avenue
New York, NY 10016-5997, USA

13 November 2006

IEEE Std 802.3as™-2006
(Amendment to
IEEE Std 802.3™-2005)

**IEEE Standard for Information technology—
Telecommunications and information exchange between systems—
Local and metropolitan area networks—
Specific requirements**

**Part 3: Carrier Sense Multiple Access with
Collision Detection (CSMA/CD) Access Method
and Physical Layer Specifications**

Amendment 3: Frame format extensions

**LAN/MAN Standards Committee
of the
IEEE Computer Society**

Approved 15 September 2006
IEEE-SA Standards Board

Abstract: This amendment to IEEE Std 802.3-2005 extends the IEEE 802.3 basic frame format with the addition of an envelope frame to accommodate additional prefixes and suffixes required by higher layer encapsulation protocols such as those defined by IEEE 802.1 (e.g., Provider Bridges, MAC Security), ITU-T or IETF (e.g., MPLS).

Keywords: 802.3as, basic frame, envelope frame, frame expansion, frame format extension, frame size, Q-tagged frame

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2006 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 13 November 2006. Printed in the United States of America.

IEEE and 802 are registered trademarks in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Incorporated.

Print: ISBN 0-7381-5229-3 SH95574
PDF: ISBN 0-7381-5230-7 SS95574

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied **“AS IS.”**

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

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

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

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

This introduction is not part of IEEE Std 802.3as-2006, IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements CSMA/CD Access Method and Physical Layer Specifications, Amendment 3: Frame format extensions.

IEEE Std 802.3 was first published in 1985. Since the initial publication, many projects have added functionality or provided maintenance updates to the specifications and text included in the standard. Each IEEE 802.3 project/amendment is identified with a suffix (e.g., IEEE Std 802.3as-2006). A historical listing of projects that have added to or modified IEEE Std 802.3 is included in IEEE Std 802.3-2005.

The media access control (MAC) protocol specified in IEEE Std 802.3 is Carrier Sense Multiple Access with Collision Detection (CSMA/CD). This MAC protocol was included in the experimental Ethernet developed at Xerox Palo Alto Research Center. While the experimental Ethernet had a 2.94 Mb/s data rate, IEEE Std 802.3-1985 specified operation at 10 Mb/s. Since 1985 new media options, new speeds of operation, and new protocol capabilities have been added to IEEE Std 802.3.

Some of the major additions to IEEE Std 802.3 are identified in the marketplace with their project number. This is most common for projects adding higher speeds of operation or new protocols. For example, IEEE Std 802.3u™ added 100 Mb/s operation (also called Fast Ethernet), IEEE Std 802.3x™ specified full duplex operation and a flow control protocol, IEEE Std 802.3z™ added 1000 Mb/s operation (also called Gigabit Ethernet), IEEE Std 802.3ae™ added 10 Gb/s operation (also called 10 Gigabit Ethernet) and IEEE Std 802.3ah™ specified access network Ethernet (also called Ethernet in the First Mile). These major additions are all now included in IEEE Std 802.3-2005 and are not maintained as separate documents.

At the date of IEEE Std 802.3as-2006 publication, IEEE Std 802.3 is comprised of the following documents:

Section One—Includes Clause 1 through Clause 20 and Annex A through Annex H and Annex 4A. Section One includes specifications for 10 Mb/s operation and the MAC, frame formats and service interfaces used for all speeds of operation.

Section Two—Includes Clause 21 through Clause 33 and Annex 22A through Annex 33E. Section Two includes management attributes for multiple protocols and speed of operation as well as specifications for providing power over twisted pair cabling for multiple operational speeds. It also includes general information on 100 Mb/s operation as well as most of the 100 Mb/s physical layer specifications.

Section Three—Includes Clause 34 through Clause 43 and Annex 36A through Annex 43C. Section Three includes general information on 1000 Mb/s operation as well as most of the 1000 Mb/s physical layer specifications. It also includes specification of IEEE 802.3 link aggregation.

Section Four—Includes Clause 44 through Clause 54 and Annex 44A through Annex 50A. Section Four includes general information on 10 Gb/s operation as well as most of the 10 Gb/s physical layer specifications.

Section Five—Includes Clause 56 through Clause 67 and Annex 58A through Annex 67A. Section Five includes subscriber access physical layers and sublayers for operation from 512 kb/s to 1000 Mb/s, and defines services and protocol elements that enable the exchange of IEEE Std 802.3 format frames between stations in a subscriber access network.

IEEE Std 802.3-2005/Cor 1-2006

This corrigendum clarifies and corrects isolation text for twisted pair Ethernet physical interfaces, including harmonization for both powered and unpowered Medium Dependent interfaces.

IEEE Std 802.3anTM-2006

This amendment includes changes to IEEE Std 802.3-2005 and adds Clause 55 and Annex 55A and Annex 55B. This amendment adds a new Physical Layer for 10 Gb/s operation over balanced twisted-pair structured cabling systems.

IEEE Std 802.3aqTM-2006

This amendment includes changes to IEEE Std 802.3-2005 and adds Clause 68. This amendment adds a new Physical Layer for 10 Gb/s operation over installed multimode fiber.

IEEE Std 802.3as-2006

This amendment includes changes to IEEE Std 802.3-2005. It extends the size of the IEEE 802.3 frame format with an envelope frame.

IEEE Std 802.3 will continue to evolve. New Ethernet capabilities are anticipated to be added within the next few years as amendments to this standard.

Conformance test methodology

An additional standard, IEEE Std 1802.3TM-2001, provides conformance test information for 10BASE-T.

Notice to users

Errata

Errata, if any, for this and all other standards can be accessed at the following URL:

<http://standards.ieee.org/reading/ieee/updates/errata/index.html>

Users are encouraged to check this URL for errata periodically.

Downloads

Portions of this standard can be downloaded from the Internet. Material include PICS tables, data tables, and code. URLs are listed in the text in the appropriate sections.

Interpretations

Current interpretations can be accessed at the following URL:

<http://standards.ieee.org/reading/ieee/interp/index.html>

Patents

Attention is called to the possibility that implementations of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents or patent applications for which a license may be required to implement an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention. A patent holder or patent applicant has filed a statement of assurance that it will grant licenses under these rights without compensation or under reasonable rates and nondiscriminatory, reasonable terms and conditions to applicants desiring to obtain such licenses. The IEEE makes no representation as to the reasonableness of rates, terms and conditions of the license agreements offered by patent holders or patent applicants. Further information may be obtained from the IEEE Standards Department.

Participants

The following individuals were members of the IEEE 802.3 working group at the beginning of the IEEE P802.3as working group ballot. Individuals may have not voted, voted for approval, disapproval or abstained on this amendment.

Robert M. Grow, *Working Group Chair*

David J. Law, *Working Group Vice Chair*

Wael William Diab, *Working Group Secretary*

Steven B. Carlson, *Working Group Executive Secretary*

Kevin Q Daines, *Chair, IEEE P802.3as Task Force*

Glenn W. Parsons, *Editor, IEEE P802.3as Task Force*

Ali Abaye	Yair Darshan	Juan Jover
John Abbott	Piers Dawe	Sanjay Kasturia
Joe Abler	Chris Di Minico	Yong Kim
Luc Adriaennsens	Thomas Dineen	Jonathan King
Puneet Agarwal	Thuyen Dinh	Paul Kish
Jan Alexander	Dan Dove	David J. Koenen
Arne Alping	John Dring	Paul Kolesar
Michael Altman	Mike Dudek	Tetsu Koyama
Stephen D. Anderson	Joseph E. Dupuis	Subi Krishnamurthy
Lewis Aronson	Herman Eiliya	Aniruddha Kundu
Joseph N. Babanezhad	George Eisler	Alan Kwentus
Chet Babla	Brian S. Ensinger	Hans Lackner
Hugh Barrass	John F. Ewen	Daun Langston
Howard Baumer	Siavash Fallahi	Jeff Lapak
Denis Beaudoin	Daniel Feldman	Michael Lawton
Michaël Beck	Alan Flatman	Michael Lerer
Alexei Beliaev	Yukihiro Fujimoto	Seyoun Lim
Mike Bennett	Justin Gaither	Tom Lindsay
Ernest E. Bergmann	Ilango S. Ganga	Martin Lobel
Sudeep Bhoja	Xiao Ming Gao	Dennis Lou
Harmeet Bhugra	Geoffrey M. Garner	Jeffrey Lynch
Brad Booth	John George	Eric R. Lynskey
Peter Bradshaw	Floyd Gerhardt	Arthur Marris
Al Braga	Ali Ghiasi	David W. Martin
Richard Brand	Joel Goergen	Thomas Mathey
Ralf-Peter Braun	Glenn Golden	Frank McCarthy
Robert D. Brink	Rich Graham	Brett McClellan
Benjamin Brown	Sandeep Gupta	Mike McConnell
Kevin Brown	Tanmay Gupta	Michael S. McCormack
Brian Brunn	Russ Gyurek	Richard Y. Mei
Robert Brunner	Bernie Hammond	Richard Mellitz
Robert Busse	Asif Hazarika	Amir Mezer
Jeff Cain	Adam Healey	Tremont Miao
Luke Chang	Gopal Hegde	Charles Moore
Kok-Wui Cheong	Tricia Hill	Wayne A. Mueller
Jae Hun Cho	Sammy Hindi	Shimon Muller
Rahul Chopra	Chuck Hudson	Jim Muth
George Claseman	Walter K. Hurwitz	Hari Naidu
Terry Cobb	John Jaeger	Shinji Nishimura
Larry Cohen	David V. James	Bob Noseworthy
Edward Cornejo	Andrew C. Jimenez	Satoshi Obara
George Cravens	Cristopher Jolly	John Oberstar
David Cunningham	Chad Jones	Mike Oltmanns
Dariusz Dabiri	William W. Jones	Tom Palkert
John Dallesasse	Seong-Soon Joo	Bill Panos
John D'Ambrosia	Thomas K. Jørgensen	Martin Patoka

Jan P. Peeters Weem
 Petar Pepeljugin
 William R. Peters
 Abhijit Phanse
 David Piete
 Rick Pimpinella
 Timothy R. Plunkett
 Petre Popescu
 Scott R. Powell
 Haoli Qian
 Rick Rabinovich
 Sailesh K. Rao
 Dan Rausch
 Pedro Reviriego
 Albrecht Rommel
 Valerie Rybinski
 Hyunsurk (Eric) Ryu
 Olindo Savi
 T. Shannon Sawyer

Brian Seemann
 Katsutoshi Seki
 Lee Sendelbach
 Abhijit Shanbhag
 Grant Smith
 Fulvio Spagna
 Bryan Sparrowhawk
 Matthew B. Squire
 David Srodzinski
 Clay Stanford
 Steve Swanson
 Norman L. Swenson
 Andre Szczepanek
 Vivek Telang
 Jose Tellado
 Patricia A. Thaler
 Geoffrey O. Thompson
 Lars E. Thon

Peter Tomaszewski
 Hidehiro Toyoda
 Matt Traverso
 Bor-Long Twu
 Sterling A. Vaden
 Magesh Valliappan
 Nicholas Van Bavel
 Schelto J. van Doorn
 Andre VanSchyndel
 Albert Vareljian
 Brian Von Herzen
 Manoj Wadekar
 Nick Weiner
 Kevin Witt
 Bill Woodruff
 Ariel Yagil
 George Zimmerman
 Pavel Zivny
 Bob Zona

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

John S. Abbott
 Eng Ahmed Abdelhalim
 Andrew I. Abolafia
 Toru Aihara
 Adewole C. Akpose
 Thomas Alexander
 Butch Anton
 Chris B. Bagge
 Hugh Barrass
 Leslie A. Baxter
 Michael A. Beck
 Alexei Beliaev
 Rahul B. Bhushan
 Gennaro Boggia
 Brad J. Booth
 Ralf-Peter Braun
 Benjamin J. Brown
 Matthew K. Burnburg
 William A. Byrd
 Sean S. Cai
 Jeffrey C. Cain
 Edward J. Carley, Jr
 James T. Carlo
 Juan C. Carreon
 Martin Carroll
 Yi-ming Chen
 Danila Chernetsov
 Keith Chow
 Tommy P. Cooper
 Kalyan R. Dharanipragada
 Wael W. Diab
 Russell S. Dietz
 Thomas J. Dineen
 Carlo Donati
 Sourav K. Dutta
 Michael A. Fischer
 Andre F. Fournier
 David Frattura
 Yukihiro Fujimoto
 Devon L. Gayle
 Michael D. Geipel

Fernando Genkuong
 Joel R. Goergen
 Sergiu R. Goma
 Ron K. Greenthaler
 Randall C. Groves
 Robert M. Grow
 C. G. Guy
 Joseph M. Gwinn
 Stephen R. Haddock
 John F. Hawkins
 Adam B. Healey
 Dennis Horwitz
 C. R. Huntley
 Guy T. Hutchison
 Atsushi Ito
 Peeya Iwagoshi
 Raj Jain
 David V. James
 Tony A. Jeffree
 Peter G. Jones
 William W. Jones
 Juan M. Jover
 Shinkyō Kaku
 Hassan Kamgar
 Piotr Karocki
 Stuart J. Kerry
 Lior Khernosh
 Yongbum Kim
 Glen Kramer
 David J. Law
 Shawn M. Leard
 William Lumpkins
 G. L. Luri
 Eric R. Lynskey
 Arthur Marris
 Richard A. McBride
 Jonathon C. McLendon
 John L. Messenger
 George J. Miao
 Gary L. Michel
 Scott Misha

William J. Mitchell
 Dinesh Mohan
 Jose Morales
 Shimon Muller
 Michael S. Newman
 Nick Nikjoo
 Satoshi Obara
 Robert O'Hara
 Barry A. O'Mahony
 Glenn W. Parsons
 Peter K. Pepeljugin
 Subburajan Ponnuswamy
 Vikram Punj
 Maximilian Riegel
 Gary S. Robinson
 Philip T. Robinson
 Robert A. Robinson
 John J. Roese
 Jon W. Rosdahl
 Randall M. Safier
 Perry L. Schwartz
 Michael J. Seaman
 Rich Seifert
 Manikantan Srinivasan
 Thomas E. Starai
 Norman L. Swenson
 Michael D. Johas Teener
 Jose Tellado
 Patricia A. Thaler
 David W. Thompson
 Geoffrey O. Thompson
 Jeffrey R. Towne
 Mark-Rene Uchida
 Scott A. Valcourt
 Schelto J. van Doorn
 Timothy P. Walker
 Frederick J. Weniger
 Robert D. Wilkens
 Takahito Yoshizawa
 Oren Yuen

When the IEEE-SA Standards Board approved this standard on 15 September 2006, it had the following membership:

Steve M. Mills, *Chair*
Richard H. Hulett, *Vice Chair*
Don Wright, *Past Chair*
Judith Gorman, *Secretary*

Mark D. Bowman
Dennis B. Brophy
William R. Goldbach
Arnold M. Greenspan
Robert M. Grow
Joanna N. Guenin
Julian Forster*
Mark S. Halpin
Kenneth S. Hanus

William B. Hopf
Joseph L. Koepfinger*
David J. Law
Daleep C. Mohla
T. W. Olsen
Glenn Parsons
Ronald C. Petersen
Tom A. Prevost

Greg Ratta
Robby Robson
Anne-Marie Sahazizian
Virginia C. Sulzberger
Malcolm V. Thaden
Richard L. Townsend
Walter Weigel
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Satish K. Aggarwal, *NRC Representative*
Richard DeBlasio, *DOE Representative*
Alan H. Cookson, *NIST Representative*

Michelle Turner
IEEE Standards Program Manager, Document Development

Michael D. Kipness
IEEE Standards Program Manager, Technical Program Development

List of special symbols

For the benefit of those who have received this document by electronic means, what follows is a list of special symbols and operators. If any of these symbols or operators fail to display correctly, the editors hope that this table will aid in interpreting any funny blobs and strokes appearing in the body of the document.

Printed Character	Meaning	Font
*	Boolean AND	Symbol
+	Boolean OR, arithmetic addition	Symbol
^	Boolean XOR	Times New Roman
!	Boolean NOT	Symbol
×	Multiplication	Symbol
<	Less than	Symbol
≤	Less than or equal to	Symbol
>	Greater than	Symbol
≥	Greater than or equal to	Symbol
=	Equal to	Symbol
≠	Not equal to	Symbol
←	Assignment operator	Symbol
∈	Indicates membership	Symbol
∉	Indicates nonmembership	Symbol
±	Plus or minus (a tolerance)	Symbol
°	Degrees	Symbol
∑	Summation	Symbol
√	Square root	Symbol
—	Big dash (em dash)	Times New Roman
–	Little dash (en dash), subtraction	Times New Roman
	Vertical bar	Times New Roman
†	Dagger	Times New Roman
‡	Double dagger	Times New Roman
α	Lower case alpha	Symbol
β	Lower case beta	Symbol
γ	Lower case gamma	Symbol
δ	Lower case delta	Symbol
ε	Lower case epsilon	Symbol
λ	Lambda	Symbol
μ	Micro	Times New Roman
Ω	Omega	Symbol

Contents

1.	Introduction.....	1
2.	Media Access Control (MAC) service specification	3
3.	Media Access Control (MAC) frame and packet specifications	5
4.	Media Access Control.....	11
	Annex 4A (normative) Simplified full duplex media access control	31
5.	Layer Management	41
30.	Management.....	45
	Annex 30A (normative) GDMO specification for IEEE 802.3 managed object classes	49
	Annex 30B (normative) GDMO and ASN.1 definitions for management	51
31.	MAC Control	53
	Annex 31B (normative) MAC Control PAUSE operation	59
43.	Link Aggregation	63
	Annex 43B (normative) Requirements for support of Slow Protocols.....	65
57.	Operations, Administration, and Maintenance (OAM)	67
61.	Physical Coding Sublayer (PCS), Transmission Convergence (TC) sublayer, and common specifications, type 10PASS-TS and type 2BASE-TL	69
64.	Multipoint MAC Control	71
65.	Extensions of the Reconciliation Sublayer (RS) and Physical Coding Sublayer (PCS)/Physical Media Attachment (PMA) for 1000BASE-X for multipoint links and forward error correction.....	89

IEEE Standard for Information technology—
Telecommunications and information exchange between systems—
Local and metropolitan area networks—
Specific requirements

Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications

Amendment 3: Frame format extensions

NOTE—The editing instructions contained in this amendment define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italic***. Four editing instructions are used: change, delete, insert, and replace. ***Change*** is used to make corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using ~~strike through~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one. Editorial notes will not be carried over into future editions because the changes will be incorporated into the base standard.

1. Introduction

1.3 Normative references

Change the introductory paragraph as follows:

The following standards contain provisions that, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties subject to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid international standards. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

Change the following normative reference:

IEEE Std 802.1QTM ~~1998~~, IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

1.4 Definitions

Change the following definitions and update alphabetical order:

1.4.127 ~~data~~ MAC frame: Consists of the Destination Address, Source Address, ~~Length Field, logical link control (LLC) Data, PAD~~Length/Type field, MAC Client Data, Pad (if required), and Frame Check Sequence.

1.4.334 ~~tagged MAC frame:~~ A frame that contains a QTag Prefix**Q-tagged frame:** A MAC frame with a specific Type value, and that has a maximum length of 1522 octets. (See IEEE 802.3, 3.2.7 and IEEE 802.1Q, Annex C)

Insert the following definitions in alphabetic order:

1.4.x basic frame: A MAC frame that carries a Length/Type field with the Length or Type interpretation and has a maximum length of 1518 octets. The basic frame is not intended to allow inclusion of additional tags (i.e., untagged) or encapsulations required by higher layer protocols. (See IEEE 802.3, 3.2.7)

1.4.y envelope frame: A MAC frame that carries a Length/Type field with the Type interpretation that may indicate additional encapsulation information within the MAC client data and has a maximum length of 2000 octets. The envelope frame is intended to allow inclusion of additional prefixes and suffixes required by higher layer encapsulation protocols. The encapsulation protocols may use up to 482 octets. (See IEEE 802.3, 3.2.7)

2. Media Access Control (MAC) service specification

Change subclause 2.1 as follows:

2.1 Scope and field of application

This clause specifies the services provided by the Media Access Control (MAC) sublayer ~~and the optional MAC control sublayer~~ to the client of the MAC (see Figure 1–1). MAC clients may include the Logical Link Control (LLC) sublayer, Bridge Relay Entity, or other users of ISO/IEC LAN International Standard MAC services (see Figure 2–1). The services are described in an abstract way and do not imply any particular implementation or any exposed interface. ~~There is not necessarily a one-to-one correspondence between the primitives and the formal procedures and interfaces described in Clause 4 and Clause 31. Other clauses in this standard may add optional protocol sublayers directly above the MAC that preserve the service interface to the MAC client. Any augmentations to the MAC client interface are specified in the relevant sublayer clause (e.g., Clause 31).~~

Replace Figure 2–1 with the following:

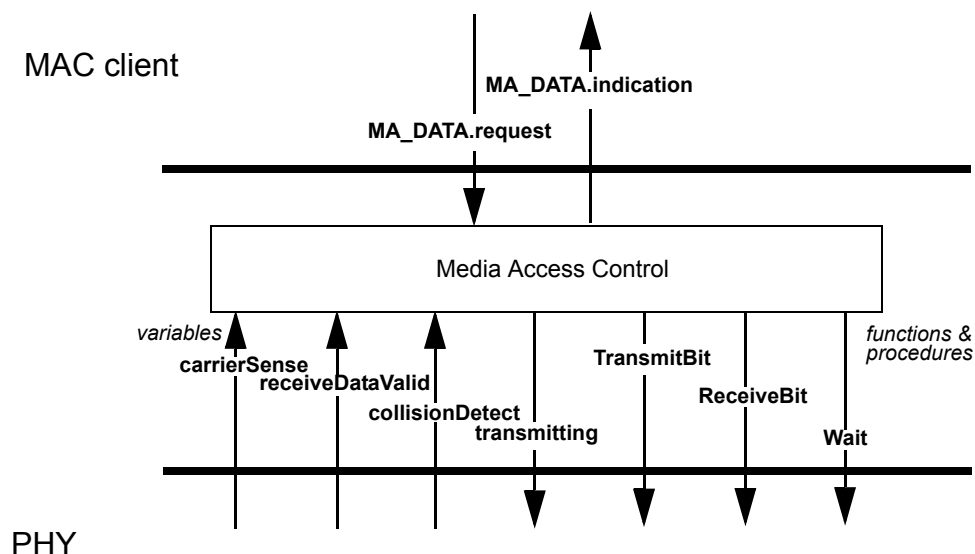


Figure 2–1—Service specification primitive relationships

Change subclause 2.2 as follows:

2.2 Overview of the service

2.2.1 General description of services provided by the layer

The services provided by the MAC sublayer allow the local MAC client entity to exchange LLC data units with peer LLC sublayer entities. Optional support may be provided for resetting the MAC sublayer entity to a known state.

~~The optional MAC control sublayer provides an additional service for controlling MAC operation. This may be used to provide flow control between peer MAC client entities across the underlying channel.~~

2.2.2 Model used for the service specification

The model used in this service specification is identical to that used in 1.2.2.

2.2.3 Overview of interactions

MA_DATA.request

MA_DATA.indication

~~MA_CONTROL.request (used by optional MAC Control sublayer)~~

~~MA_CONTROL.indication (used by optional MAC Control sublayer)~~

2.2.4 Basic services ~~and options~~

The MA_DATA.request and MA_DATA.indication service primitives described in this subclause are mandatory. ~~The MA_CONTROL.request and MA_CONTROL.indication service primitives are mandatory if the optional MAC Control sublayer is implemented.~~

Delete subclause 2.3.3 in its entirety.

Delete subclause 2.3.4 in its entirety.

Change title of Clause 3 as follows:

3. Media access control frame structureMedia Access Control (MAC) frame and packet specifications

Change subclause 3.1 as follows:

3.1 Overview

~~This clause defines in detail the frame structure for data communication systems using the CSMA/CD MAC. It defines the syntax and semantics of the various components of the MAC frame.~~

~~Two frame formats, are specified in this clause:~~

- ~~a) A basic MAC frame format, and~~
- ~~b) An extension of the basic MAC frame format for Tagged MAC frames, i.e., frames that carry QTag Prefixes.~~

This clause defines the mapping between MAC service interface primitives and Ethernet packets, including the syntax and semantics of the various fields of MAC frames and the fields used to form those MAC frames into packets.

During Ethernet's history, capabilities have been added to allow data link layer (layer 2) protocol encapsulations within the MAC Client Data field. As a result, there are now more than one type of MAC frame.

The frame format specified in this clause includes the following three types of MAC frames:

- a) A basic frame
- b) A Q-tagged frame
- c) An envelope frame

All three frame types use the same Ethernet frame format.

3.1.1 MAC frame Packet format

Figure 3–1 shows the ~~nine~~ fields of a ~~framepacket~~: the ~~preamble~~Preamble, Start Frame Delimiter (SFD), the addresses of the ~~MAC frame's source and destination~~ and source, a length or type field to indicate the length or protocol type of the following field that contains the MAC Client data, a field that contains padding if required, ~~and the frame check sequence~~Frame Check Sequence (FCS) field containing a cyclic redundancy check value to detect errors in a received MAC frame. ~~An Extension field is added, and an extension field if required (for 1000 Mb/s half duplex operation only).~~ Of these ~~nine~~ fields, all are of fixed size except for the ~~MAC Client Data, Pad and Extension~~data, pad, and extension fields, which may contain an integer number of octets between the minimum and maximum values that are determined by the specific implementation of the ~~CSMA/CD MAC~~. See 4.4 for particular ~~implementation~~ MAC parameters.

The minimum and maximum MAC frame size limits in 4.4 refer to that portion of the ~~frame packet~~ from the ~~destination address~~Destination Address field through the ~~frame check sequence~~Frame Check Sequence field, inclusive (i.e., the MAC frame).

Relative to Figure 3–1, the octets of a ~~framepacket~~ are transmitted from top to bottom, and the bits of each octet are transmitted from left to right.

Replace Figure 3–1 with the following:

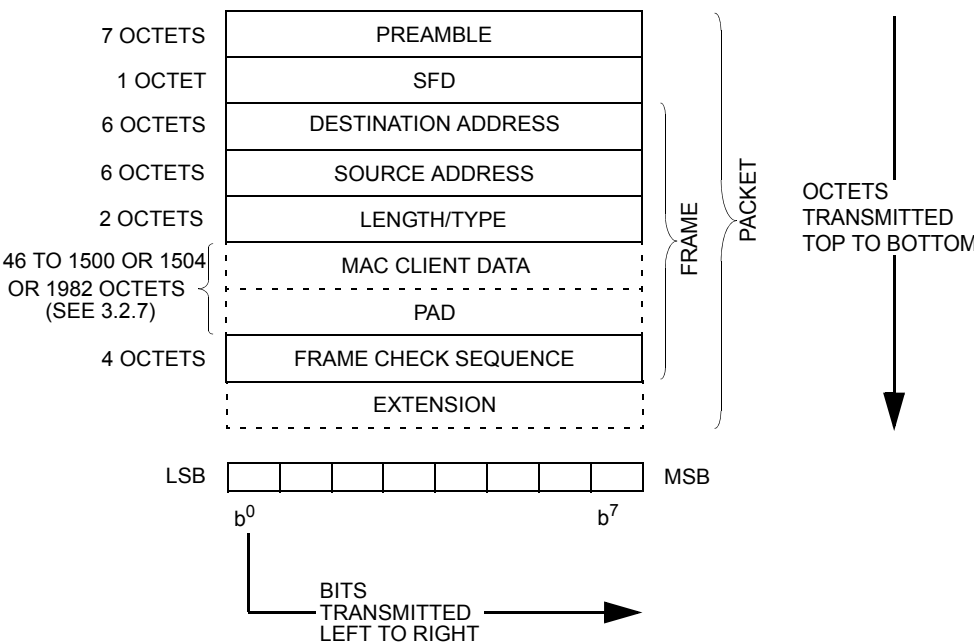


Figure 3–1—Packet format

Insert new subclause 3.1.2 as follows:

3.1.2 Service interface mappings

Figure 3–2 shows the mapping of service interface parameters to the fields of a MAC frame within a packet. The MAC client may or may not supply Pad and FCS. For this reason the mappings for Pad and FCS are shown with dashed lines.

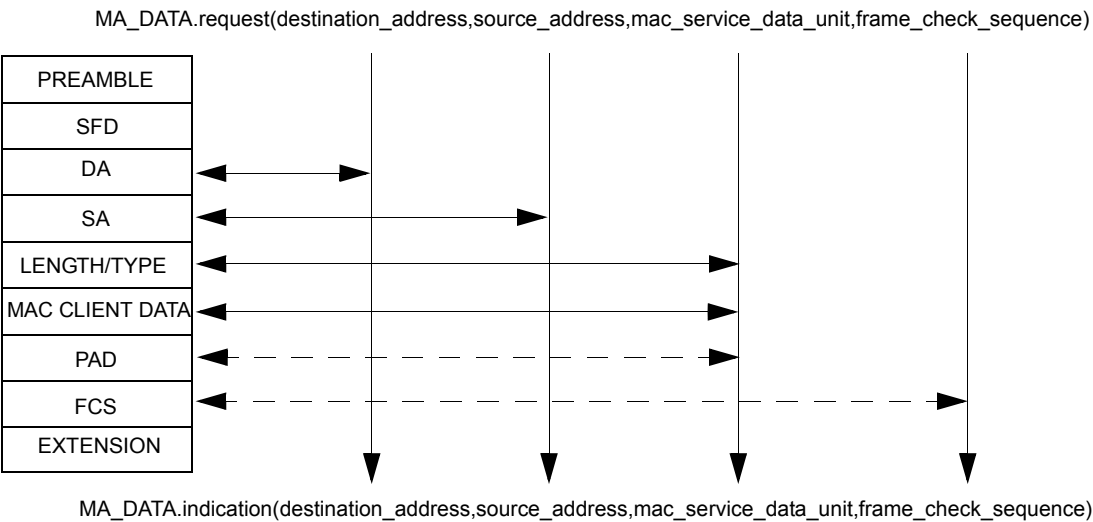


Figure 3–2—Service primitive mappings

Change subclause 3.2 as follows:

3.2 Elements of the MAC frame and packet

A MAC frame is encapsulated in a packet by the MAC. This section describes in detail the fields of the MAC frame and the additional fields that the MAC creates to encapsulate the MAC frame. These fields are described in order of transmission.

Change subclause 3.2.1 as follows:

3.2.1 Preamble field

The ~~preamble~~Preamble field is a 7-octet field that is used to allow the PLS circuitry to reach its steady-state synchronization with the received ~~frame~~packet's timing (see 4.2.5).

Change subclause 3.2.2 as follows:

3.2.2 Start Frame Delimiter (SFD) field

The SFD field is the sequence 10101011. It immediately follows the preamble pattern. A MAC frame starts immediately after the SFD, and indicates the start of a frame.

Change the first paragraph of subclause 3.2.3 as follows:

3.2.3 Address fields

Each MAC frame shall contain two address fields: the Destination Address field and the Source Address field, in that order. The Destination Address field shall specify the destination addressee(s) for which the MAC frame is intended. The Source Address field shall identify the station from which the MAC frame was initiated. The representation of each address field shall be as follows (see Figure 3–2):

Change subclause 3.2.4 as follows:

3.2.4 Destination Address field

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

Change subclause 3.2.5 as follows:

3.2.5 Source Address field

The Source Address field specifies the station sending the MAC frame. The Source Address field is not interpreted by the ~~CSMA/CD~~ MAC sublayer.

Change subclause 3.2.6 as follows:

3.2.6 Length/Type field

This two-octet field takes one of two meanings, depending on its numeric value. For numerical evaluation, the first octet is the most significant octet of this field.

- a) If the value of this field is less than or equal to ~~the value of maxValidFrame (as specified in 4.2.7.4)~~1500 decimal (05DC hexadecimal), then the Length/Type field indicates the number of

MAC client data octets contained in the subsequent MAC Client Data field of the basic frame (Length interpretation).

- b) If the value of this field is greater than or equal to 1536 decimal (equal to 0600 hexadecimal), then the Length/Type field indicates the nature of the MAC client protocol (Type interpretation).¹⁹ The Length and Type interpretations of this field are mutually exclusive.

When used as a Type field, it is the responsibility of the MAC client to ensure that the MAC client operates properly when the MAC sublayer pads the supplied MAC client data, as discussed in 3.2.7.

Regardless of the interpretation of the Length/Type field, if the length of the MAC Client Data field is less than the minimum required for proper operation of the protocol, a PAD field (a sequence of octets) will be added ~~at the end of~~ after the MAC Client Data field but prior to the FCS field, specified below. The procedure that determines the size of the PAD field is specified in 4.2.8. The Length/Type field is transmitted and received with the high order octet first.

NOTE—Clause 42 of IEEE Std 802a-2003 (an amendment to IEEE Std 802) defines a set of Type values and associated mechanisms for use in prototype and vendor-specific protocol development.

Change subclause 3.2.7 as follows:

3.2.7 MAC Client Data field and PAD fields

The MAC Client Data field contains a sequence of ~~n~~ octets. Full data transparency is provided in the sense that any arbitrary sequence of octet values may appear in the MAC Client Data field up to a maximum field length ~~number specified~~ determined by the particular implementation of the standard that is used. ~~A minimum frame size is required for correct CSMA/CD protocol operation and is specified by the particular implementation of the standard. If necessary, the data field is extended by appending extra bits (that is, a pad) in units of octets after the data field but prior to calculating and appending the FCS. The size of the pad, if any, is determined by the size of the data field supplied by the MAC client and the minimum frame size and address size parameters of the particular implementation. The maximum size of the data field is determined by the maximum frame size and address size parameters of the particular implementation.~~

Ethernet implementations shall support at least one of three maximum MAC Client Data field sizes defined as follows:

- a) 1500 decimal—basic frames (see 1.4.x)
- b) 1504 decimal—Q-tagged frames (see 1.4.344)
- c) 1982 decimal—envelope frames (see 1.4.y)

If layer management is implemented, frames with a MAC Client Data field larger than the supported maximum MAC Client Data field size are counted. It is recommended that new implementations support the transmission and reception of envelope frames, item c) above.

NOTE 1 —The envelope frame is intended to allow inclusion of additional prefixes and suffixes required by higher layer encapsulation protocols (see 1.4.x) such as those defined by the IEEE 802.1 working group (such as Provider Bridges and MAC Security), ITU-T or IETF (such as MPLS). The original MAC Client Data field maximum remains 1500 octets while the encapsulation protocols may add up to an additional 482 octets. Use of these extra octets for other purposes is not recommended, and may result in MAC frames being dropped or corrupted as they may violate maximum MAC frame size restrictions if encapsulation protocols are required to operate on them.

NOTE 2 —All IEEE 802.3 MAC frames share a common format. The processing of the three types of MAC frames is not differentiated within the IEEE 802.3 MAC, except for management. However, they may be distinguished within the MAC client.

¹⁹Type field assignments are administered by the Registration Authority, IEEE Standards Department (see Footnote 18 for address):
P.O. Box 1331, 445 Hoes Lane, Piscataway, NJ 08855-1331, USA; +1 732 562 3813; fax +1 732 562 1571.
URL: <http://standards.ieee.org/>.

NOTE 3 — All Q-tagged frames are envelope frames, but not all envelope frames are Q-tagged frames.

~~The length of PAD field required for MAC client data that is N octets long is $\max[0, \text{minFrameSize} - (8 \times n + 2 \times \text{addressSize} + 48)]$ bits. The maximum possible size of the data field is $\text{maxUntaggedFrameSize} - (2 \times \text{addressSize} + 48)/8$ octets. See 4.4 for a discussion of implementation MAC parameters; see 4.2.3.3 for a discussion of the minimum frame size, minFrameSize.~~

Insert a new subclause 3.2.8 as follows; renumber existing 3.2.8 and 3.2.9 to be 3.2.9 and 3.2.10 respectively:

3.2.8 Pad field

A minimum MAC frame size is required for correct CSMA/CD protocol operation (see 4.2.3.3 and 4.4). If necessary, a Pad field (in units of octets) is appended after the MAC Client Data field prior to calculating and appending the FCS field. The size of the Pad, if any, is determined by the size of the MAC Client Data field supplied by the MAC client and the minimum MAC_frame size and address size MAC parameters (see 4.4).

The length of the Pad field required for MAC Client Data that is clientDatasize/8 octets long is $\max[0, \text{minFrameSize} - (\text{clientDatasize} + 2 \times \text{addressSize} + 48)]$ bits.

Change 3.2.8, renumbered to 3.2.9, as follows:

3.2.9 Frame Check Sequence (FCS) field

A cyclic redundancy check (CRC) is used by the transmit and receive algorithms to generate a CRC value for the FCS field. ~~The frame check sequence (FCS) field contains a 4-octet (32-bit) cyclic redundancy check (CRC) value. This value is computed as a function of the contents of the protected fields of the MAC frame: the Destination Address, Source Address, source address, destination address, Length/Type field, LLC MAC Client Data and Pad data and pad (that is, all fields except the preamble, SFD, FCS, and extension).~~ The encoding is defined by the following generating polynomial.

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

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

- The first 32 bits of the MAC frame are complemented.
- The n bits of the ~~protected fields~~frame are then considered to be the coefficients of a polynomial $M(x)$ of degree $n - 1$. (The first bit of the Destination Address field corresponds to the $x^{(n-1)}$ term and the last bit of the MAC Client Data~~data~~ field (or Pad field if present) corresponds to the x^0 term.)
- $M(x)$ is multiplied by x^{32} and divided by $G(x)$, producing a remainder $R(x)$ of degree ≤ 31 .
- The coefficients of $R(x)$ are considered to be a 32-bit sequence.
- The bit sequence is complemented and the result is the CRC.

The 32 bits of the CRC value are placed in the ~~frame check sequence~~FCS field so that the x^{31} term is the left-most bit of the first octet, and the x^0 term is the right most bit of the last octet. (The bits of the CRC are thus transmitted in the order $x^{31}, x^{30}, \dots, x^1, x^0$.) See Hammond, et al. [B32].

Delete subclause 3.5 including Figure 3-3.

4. Media Access Control

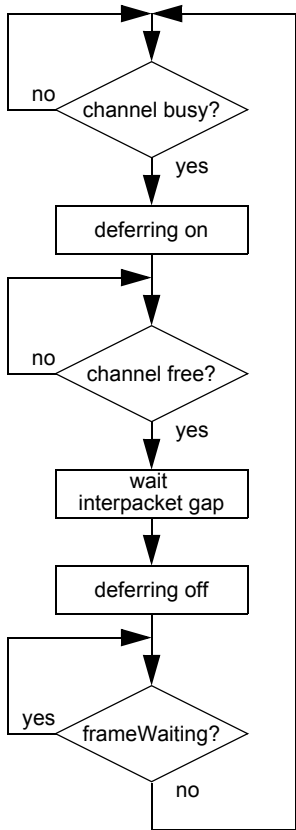
Change subclause 4.2.3 as follows:

4.2.3 ~~Frame~~Packet transmission model

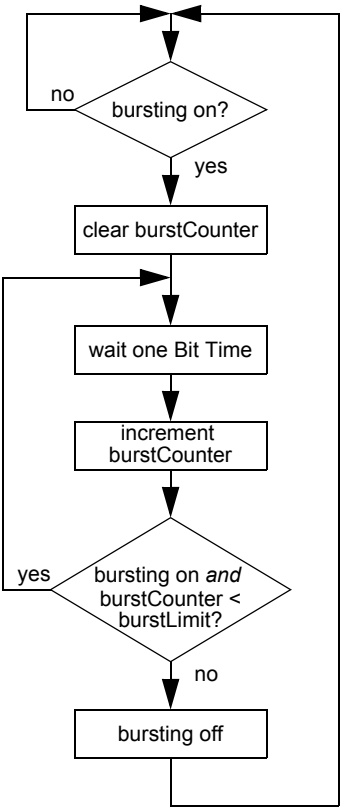
~~Frame~~Packet transmission includes the following data encapsulation and Media Access management aspects:

- a) Transmit Data Encapsulation includes the assembly of the outgoing ~~frame~~packet (from the values provided by the MAC client) and frame check sequence generation (if not provided by the MAC client).
- b) Transmit Media Access Management includes carrier deference, ~~interframe spacing~~interpacket gap, collision detection and enforcement, collision backoff and retransmission, carrier extension and ~~frame~~packet bursting.

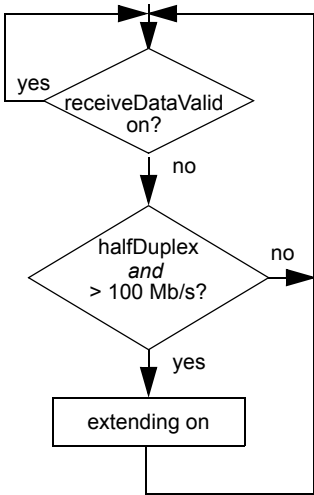
Replace Figure 4-3b in 4.2.3.1 with the following:



Deference process



*BurstTimer process



*SetExtending process

*Applicable only to half duplex operation at 1000 Mb/s

b) MAC sublayer

Figure 4–3b—Control flow

Change subclause 4.2.3.2 as follows:

4.2.3.2 Transmit media access management

4.2.3.2.1 Deference

When a ~~framepacket~~ is submitted by the MAC client for transmission, the transmission is initiated as soon as possible, but in conformance with the rules of deference stated below. The rules of deference differ between half duplex and full duplex modes.

a) *Half duplex mode*

Even when it has nothing to transmit, the CSMA/CD MAC sublayer monitors the physical medium for traffic by watching the carrierSense signal provided by the PLS. Whenever the medium is busy, the CSMA/CD MAC defers to the passing ~~framepacket~~ by delaying any pending transmission of its own. After the last bit of the passing ~~framepacket~~ (that is, when *carrierSense* changes from true to false), the CSMA/CD MAC continues to defer for a proper ~~interFrameSpacinginterPacketGap~~ (see 4.2.3.2.2).

If, at the end of the ~~interFrameSpacinginterPacketGap~~, a ~~framepacket~~ is waiting to be transmitted, transmission is initiated independent of the value of carrierSense. When transmission has completed (or immediately, if there was nothing to transmit) the CSMA/CD MAC sublayer resumes its original monitoring of carrierSense.

NOTE—It is possible for the PLS carrier sense indication to fail to be asserted briefly during a collision on the media. If the Deference process simply times the ~~interframeinterpacket~~ gap based on this indication it is possible for a short ~~interframeinterpacket~~ gap to be generated, leading to a potential reception failure of a subsequent ~~framepacket~~. To enhance system robustness the following optional measures, as specified in 4.2.8, are recommended when ~~interFrameSpacingPart1interPacketGapPart1~~ is other than zero:

Start the timing of the ~~interFrameSpacinginterPacketGap~~ as soon as transmitting and carrierSense are both false. Reset the ~~interFrameSpacinginterPacketGap~~ timer if carrierSense becomes true during the first 2/3 of the ~~interFrameSpacinginterPacketGap~~ timing interval. During the final 1/3 of the interval, the timer shall not be reset to ensure fair access to the medium. An initial period shorter than 2/3 of the interval is permissible including zero.

b) *Full duplex mode*

In full duplex mode, the CSMA/CD MAC does not defer pending transmissions based on the carrierSense signal from the PLS. Instead, it uses the internal variable *transmitting* to maintain proper MAC state while the transmission is in progress. After the last bit of a transmitted frame, (that is, when *transmitting* changes from true to false), the MAC continues to defer for a proper ~~interFrameSpacinginterPacketGap~~ (see 4.2.3.2.2).

4.2.3.2.2 ~~Interpacket gapInterframe spacing~~

As defined in 4.2.3.2.1, the rules for deferring to passing ~~framespackets~~ ensure a minimum ~~interframeinterpacket~~ spacing of ~~interFrameSpacinginterPacketGap~~ bit times. This is intended to provide ~~interframeinterpacket~~ recovery time for other CSMA/CD sublayers and for the physical medium.

Note that ~~interFrameSpacinginterPacketGap~~ is the minimum value of the ~~interframe-spacinginterpacket gap~~. If necessary for implementation reasons, a transmitting sublayer may use a larger value with a resulting decrease in its throughput. The larger value is determined by the parameters of the implementation, see 4.4.

A larger value for ~~interframe-spacinginterpacket gap~~ is used for dynamically adapting the nominal data rate of the MAC sublayer to SONET/SDH data rates (with packet granularity) for WAN-compatible applications of this standard. While in this optional mode of operation, the MAC sublayer counts the number of bits sent

during a frame's transmission. After the framepacket's transmission has been completed, the MAC sublayer extends the minimum interframe spacinginterpacket gap by a number of bits that is proportional to the length of the previously transmitted framepacket. For more details, see 4.2.7 and 4.2.8.

4.2.3.2.3 Collision handling (half duplex mode only)

Once a CSMA/CD sublayer has finished deferring and has started transmission, it is still possible for it to experience contention for the medium. Collisions can occur until acquisition of the network has been accomplished through the deference of all other stations' CSMA/CD sublayers.

The dynamics of collision handling are largely determined by a single parameter called the slot time. This single parameter describes three important aspects of collision handling:

- a) It is an upper bound on the acquisition time of the medium.
- b) It is an upper bound on the length of a framepacket fragment generated by a collision.
- c) It is the scheduling quantum for retransmission.

To fulfill all three functions, the slot time shall be larger than the sum of the Physical Layer round-trip propagation time and the Media Access Layer maximum jam time. The slot time is determined by the parameters of the implementation, see 4.4.

4.2.3.2.4 Collision detection and enforcement (half duplex mode only)

Collisions are detected by monitoring the collisionDetect signal provided by the Physical Layer. When a collision is detected during a framepacket transmission, the transmission is not terminated immediately. Instead, the transmission continues until additional bits specified by jamSize have been transmitted (counting from the time collisionDetect went on). This collision enforcement or jam guarantees that the duration of the collision is sufficient to ensure its detection by all transmitting stations on the network. The content of the jam is unspecified; it may be any fixed or variable pattern convenient to the Media Access implementation; however, the implementation shall not be intentionally designed to be the 32-bit CRC value corresponding to the (partial) framepacket transmitted prior to the jam.

4.2.3.2.5 Collision backoff and retransmission (half duplex mode only)

When a transmission attempt has terminated due to a collision, it is retried by the transmitting CSMA/CD sublayer until either it is successful or a maximum number of attempts (attemptLimit) have been made and all have terminated due to collisions. Note that all attempts to transmit a given framepacket are completed before any subsequent outgoing framepackets are transmitted. The scheduling of the retransmissions is determined by a controlled randomization process called "truncated binary exponential backoff." At the end of enforcing a collision (jamming), the CSMA/CD sublayer delays before attempting to retransmit the framepacket. The delay is an integer multiple of slotTime. The number of slot times to delay before the n th retransmission attempt is chosen as a uniformly distributed random integer r in the range:

$$0 \leq r < 2^k$$

where

$$k = \min(n, 10)$$

If all attemptLimit attempts fail, this event is reported as an error. Algorithms used to generate the integer r should be designed to minimize the correlation between the numbers generated by any two stations at any given time.

Note that the values given above define the most aggressive behavior that a station may exhibit in attempting to retransmit after a collision. In the course of implementing the retransmission scheduling procedure, a station may introduce extra delays that will degrade its own throughput, but in no case may a station's retransmission scheduling result in a lower average delay between retransmission attempts than the procedure defined above.

4.2.3.2.6 Full duplex transmission

In full duplex mode, there is never contention for a shared physical medium. The Physical Layer may indicate to the MAC that there are simultaneous transmissions by both stations, but since these transmissions do not interfere with each other, a MAC operating in full duplex mode must not react to such Physical Layer indications. Full duplex stations do not defer to received traffic, nor abort transmission, jam, backoff, and reschedule transmissions as part of Transmit Media Access Management. Transmissions may be initiated whenever the station has a framepacket queued, subject only to the interframe spacinginterpacket gap required to allow recovery for other sublayers and for the physical medium.

4.2.3.2.7 FramePacket bursting (half duplex mode only)

At an operating speed of 1000 Mb/s, an implementation may optionally transmit a series of framepackets without relinquishing control of the transmission medium. This mode of operation is referred to as *burst mode*. Once a framepacket has been successfully transmitted, the transmitting station can begin transmission of another framepacket without contending for the medium because all of the other stations on the network will continue to defer to its transmission, provided that it does not allow the medium to assume an idle condition between framepackets. The transmitting station fills the interframe spacinginterpacket gap interval with extension bits, which are readily distinguished from data bits at the receiving stations, and which maintain the detection of carrier in the receiving stations. The transmitting station is allowed to initiate framepacket transmission until a specified limit, referred to as *burstLimit*, is reached. The value of *burstLimit* is specified in 4.4.2. Figure 4–5 shows an example of transmission with framepacket bursting.

The first framepacket of a burst will be extended, if necessary, as described in 4.2.3.4. Subsequent framepackets within a burst do not require extension. In a properly configured network, and in the absence of errors, collisions cannot occur during a burst at any time after the first framepacket of a burst (including any extension) has been transmitted. Therefore, the MAC will treat any collision that occurs after the first framepacket of a burst, or that occurs after the *slotTime* has been reached in the first framepacket of a burst, as a late collision.

Replace Figure 4–5 with the following:

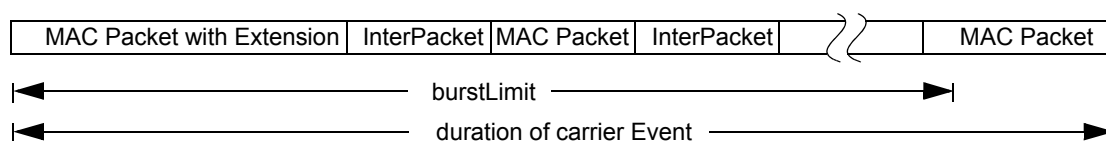


Figure 4–5—Packet bursting

Change subclause 4.2.3.3 as follows:

4.2.3.3 Minimum frame size

The CSMA/CD Media Access mechanism requires that a minimum frame length of minFrameSize bits be transmitted. If frameSize is less than minFrameSize, then the CSMA/CD MAC sublayer shall append extra bits in units of octets (~~padPad~~), after the end of the MAC Client Data field but prior to calculating and appending the FCS (if not provided by the MAC client). The number of extra bits shall be sufficient to ensure that the frame, from the DA field through the FCS field inclusive, is at least minFrameSize bits. If the FCS is (optionally) provided by the MAC client, the ~~padPad~~ shall also be provided by the MAC client. The content of the ~~padPad~~ is unspecified.

Change subclause 4.2.4.2.1 as follows:

4.2.4.2.1 Framing

The CSMA/CD sublayer recognizes the boundaries of an incoming MAC frame by monitoring the receiveDataValid signal provided by the Physical Layer. Two possible length errors can occur that indicate ill-framed data: the MAC frame may be too long, or its length may not be an integer number of octets.

- a) *Maximum Frame Size.* The receiving CSMA/CD sublayer is not required to enforce the MAC frame size limit, but it is allowed to truncate MAC frames longer than ~~maxUntaggedFrameSize~~ maxFrameSizeLimit octets (see 4.2.7.1). ~~If optional layer management is implemented, such frames may be counted whether or not they are truncated. They may also be reported as an implementation-dependent error and report this event as an (implementation-dependent) error. A receiving CSMA/CD sublayer that supports tagged MAC frames (see 3.5) may similarly truncate frames longer than (maxUntaggedFrameSize + qTagPrefixSize) octets in length, and report this event as an (implementation-dependent) error.~~

CAUTION

It is recommended that any implementation that truncates MAC frames should invalidate those frames as they may have severely weakened error protection and may cause serious problems if forwarded to the MAC client.

- b) *Integer Number of Octets in Frame.* Since the format of a valid MAC frame specifies an integer number of octets, only a collision or an error can produce a MAC frame with a length that is not an integer multiple of 8 bits. Complete MAC frames (that is, not rejected as collision fragments; see 4.2.4.2.2) that do not contain an integer number of octets are truncated to the nearest octet boundary. If frame check sequence validation detects an error in such a MAC frame, the status code alignmentError is reported.

When a burst of MAC frames is received while operating in half duplex mode at an operating speed of 1000 Mb/s, the individual MAC frames within the burst are delimited by sequences of ~~interframe~~ interpacket fill symbols, which are conveyed to the receiving MAC sublayer as extension bits. Once the collision filtering requirements for a given MAC frame, as described in 4.2.4.2.2, have been satisfied, the receipt of an extension bit can be used as an indication that all of the data bits of the MAC frame have been received.

Change subclause 4.2.7.1 as follows:

4.2.7.1 Common constants, types, and variables

The following declarations of constants, types and variables are used by the MAC frame transmission and reception sections of each CSMA/CD sublayer:

```

const
  addressSize = 48; {In bits, in compliance with 3.2.3}
  lengthOrTypeSize = 16; {In bits}
  clientDataSize = ...; {In bits, size of MAC client data Client Data; see 4.2.2.2, a) 3)}
  padSize = ...; {In bits, = max (0, minFrameSize — (2 × addressSize + lengthOrTypeSize +
                                clientDataSize + crcSize))}
  dataSize = ...; {In bits, = clientDataSize + padSize}
  crcSize = 32; {In bits, 32-bit CRC}
  frameSize = ...; {In bits, = 2 × addressSize + lengthOrTypeSize + dataSize + crcSize; see 4.2.2.2, a)}
  minFrameSize = ...; {In bits, implementation-dependent, see 4.4}
  maxUntaggedFrameSize = ...; {In octets, implementation-dependent, see 4.4}
  maxBasicFrameSize = 1518; {In octets, see 3.2.7, 4.4}
  maxEnvelopeFrameSize = 2000; {In octets, see 3.2.7, 4.4}
  qTagPrefixSize = 4; {In octets, length of QTag Prefix Q-tag prefix, see 3.5.3.2.7, 4.4}
  maxFrameSizeLimit = maxBasicFrameSize or (maxBasicFrameSize + qTagPrefixSize) or
    maxEnvelopeFrameSize; {in octets}
  extend = ...; {Boolean, true if (slotTime — minFrameSize) > 0, false otherwise}
  extensionBit = ...; {A non-data value which is used for carrier extension and interframeinterpacket
    during bursts}
  extensionErrorBit = ...; {A non-data value which is used to jam during carrier extension}
  minTypeValue = 1536; {Minimum value of the Length/Type field for Type interpretation}
  maxValidFrame = maxUntaggedFrameSize — (2 × addressSize + lengthOrTypeSize + crcSize) / 8;
  {In octets, the maximum length of the MAC client data field. This constant is
  defined for editorial convenience, as a function of other constants}
  maxBasicDataSize = 1500;
    {In octets, the maximum length of the MAC Client Data field of the basic frame.}
  slotTime = ...; {In bit times, unit of time for collision handling, implementation-dependent, see 4.4}
  preambleSize = 56; {In bits, see 4.2.5}
  sfdSize = 8; {In bits, start frame delimiter Start Frame Delimiter}
  headerSize = 64; {In bits, sum of preambleSize and sfdSize}

type
  Bit = (0, 1);
  PhysicalBit = (0, 1, extensionBit, extensionErrorBit);
    {Bits transmitted to the Physical Layer can be either 0, 1, extensionBit or
    extensionErrorBit. Bits received from the Physical Layer can be either 0, 1
    or extensionBit}
  AddressValue = array [1..addressSize] of Bit;
  LengthOrTypeValue = array [1..lengthOrTypeSize] of Bit;
  DataValue = array [1..dataSize] of Bit; {Contains the portion of the MAC frame that starts with the first
    bit following the Length/Type field and ends with the last bit
    prior to the FCS field. For VLAN Tagged frames, this value
    includes the Tag Control Information field and the original
    MAC client Length/Type field. See 3.5}
  CRCValue = array [1..crcSize] of Bit;
  PreambleValue = array [1..preambleSize] of Bit;
  SfdValue = array [1..sfdSize] of Bit;

```