Copies of this document may be purchased from:
Global Engineering, 15 Inverness Way East,
Englewood, CO 80112-5704
Phone: (800) 854-7179 or (303) 792-2181 Fax: (303) 792-2192

X3.297-1996
X3T11/Project 901D/Rev 7.4

FIBRE CHANNEL

PHYSICAL AND SIGNALLING INTERFACE - 2 (FC-PH-2) Rev 7.4

working draft proposed
American National Standard
for Information Systems

September 10, 1996

Secretariat:
Computer & Business Equipment Manufacturers Association

ABSTRACT: This standard describes the enhancement to ANSI X3.230, Fibre Channel Physical and Signalling Interface (FC-PH).

NOTE:

This is an internal working document of X3T11, a Technical Committee of Accredited Standards Committee X3. As such, this is not a completed standard. The contents are actively being modified by the X3T11 Task Group. This document is made available for review and comment only. For current information on the status of this document contact the individuals shown below:

POINTS OF CONTACT:

Roger Cummings (Chairman)
Distributed Processing Technology
140 Candace Drive
Maitland, FL 32751
(407) 830-5522 x348 Fax: (407) 260-5366
E-Mail: cummings_roger@dpt.com

I. Dal Allan
(Fibre Channel Working Group Chairman)
ENDL
14426 Black Walnut Court
Saratoga, CA 95070
(408) 867-6630 Fax: (408) 867-2115
E-Mail: dal.allan@mcimail.com

Carl Zeitler (Vice Chairman)
IBM Corporation, MS 9250
11400 Burnet Road, Austin, TX 78758
(512) 838-1797 Fax: (512) 838-3703
E-Mail: zeitler@ausvm6.vnet.ibm.com

K. C. Chennappan (Technical Editor)
IBM Corporation, MS 9250
11400 Burnet Road, Austin, TX 78758
(512) 838-1798 Fax: (512) 838-3703
E-Mail: kcc@vnet.ibm.com

ANSI ®
dp ANS X3.297-1996

draft proposed American National Standard
for Information Systems –

Fibre Channel – Physical and Signalling Interface - 2

Secretariat

Information Technology Industry Council

Approved

American National Standards Institute, Inc.

Abstract

This standard describes the enhancement to ANSI X3.230, Fibre Channel Physical and Signalling Interface (FC-PH).

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer.

Consensus is established when, in the judgement of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made towards their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give interpretation on any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

Published by

American National Standards Institute
11 W. 42nd Street, New York, New York 10036

Contents

Annexes

TABLES

Figures

Foreword (This Foreword is not part of American National Standard X3.297-1996.)

This Fibre Channel, Physical and Signalling Interface - 2 standard (FC-PH-2) describes the enhancement to ANSI X3.230 FC-PH.

This standard was developed by Task Group X3T11 of Accredited Standards Committee X3 during 1994. The standards approval process started in 19xx. This document includes annexes which are informative and are not considered part of the standard.

Requests for interpretation, suggestions for improvement or addenda, or defect reports are welcome. They should be sent to the X3 Secretariat, Computer and Business Equipment Manufactures Association, 1250 Eye Street, NW, Suite 200, Washington, DC 20005.

This standard was processed and approved for submittal to ANSI by Accredited Standard Committee on Information Processing Systems, X3. Committee approval of the standard does not necessarily imply that all committee members voted for approval. At the time it approved this standard, the X3 Committee had the following members:

NOTE - The developers of this standard have requested that holders of patents that may be required for the implementation of the standard, disclose such patents to the publisher. However neither the developers nor the publisher have undertaken a patent search in order to identify which if any patents may apply to this standard. No position is taken with respect to the validity of any claim or any patent rights that may have been disclosed. Details may be obtained from the publisher concerning any statement of patents and willingness to grant a license on a nondiscriminatory basis and with reasonable terms and conditions to applicants desiring to obtain such a license.

GUIDE International	Frank Kirshenbaum
	Harold Kuneke (Alt.)
Hewlett-Packard	Donald C. Loughry
Hitachi America Ltd.	Kei Yamashita
	John Neumann (Alt.)
Hughes Aircraft Company	Harold L. Zebrack
IBM Corporation	Robert H. Follett
	Mary Anne Gray (Alt.)
Lawrence Berkeley Laboratory	David F. Stevens
	Robert L. Fink (Alt.)
National Communications Systems	Dennis Bodson
	George W. White (Alt.)
National Institute of Standards and Technology	Robert E. Rountree
	Michael D. Hogan (Alt.)
NCR Corporation	Thomas W. Kern
	A. R. Daniels (Alt.)
OMNICOM, Inc.	Harold C. Folts
	Cheryl C. Slobodian (Alt.)
Open Software Foundation	Fritz Schulz
Recognition Technology Users Association	Herbert F. Schantz
	G. Edwin Hale (Alt.)
SHARE, Inc.	Thomas B. Steel Jr.
	Gary Ainsworth (Alt.)
Sony Corporation	Michael Deese
Storage Technology Corporation	Joseph S. Zajaczkowski
	Samuel Cheatham (Alt.)
Sun Microsystems, Inc.	Scott K. Jameson
3M Company	Paul D. Jahnke
Unisys Corporation	Stephen P. Oksala
	John L. Hill (Alt.)
U. S. Department of Defense	William C. Rinehuls
	Thomas E. Bozek (Alt.)
	Fred Virtue (Alt.)
U. S. General Services Administration	Douglas K. Arai
	Larry L. Jackson (Alt.)
US West Corporation	Gary Dempsey
	Susan Capraro (Alt.)
USE, Inc.	Pete Epstein
Wang Laboratories, Inc.	Steve Brody
	Sarah Wagner (Alt.)
Wintergreen Information Services	John L. Wheeler
Xerox Corporation	Roy Pierce

Technical committee X3T11 which developed this standard, had the following participants:

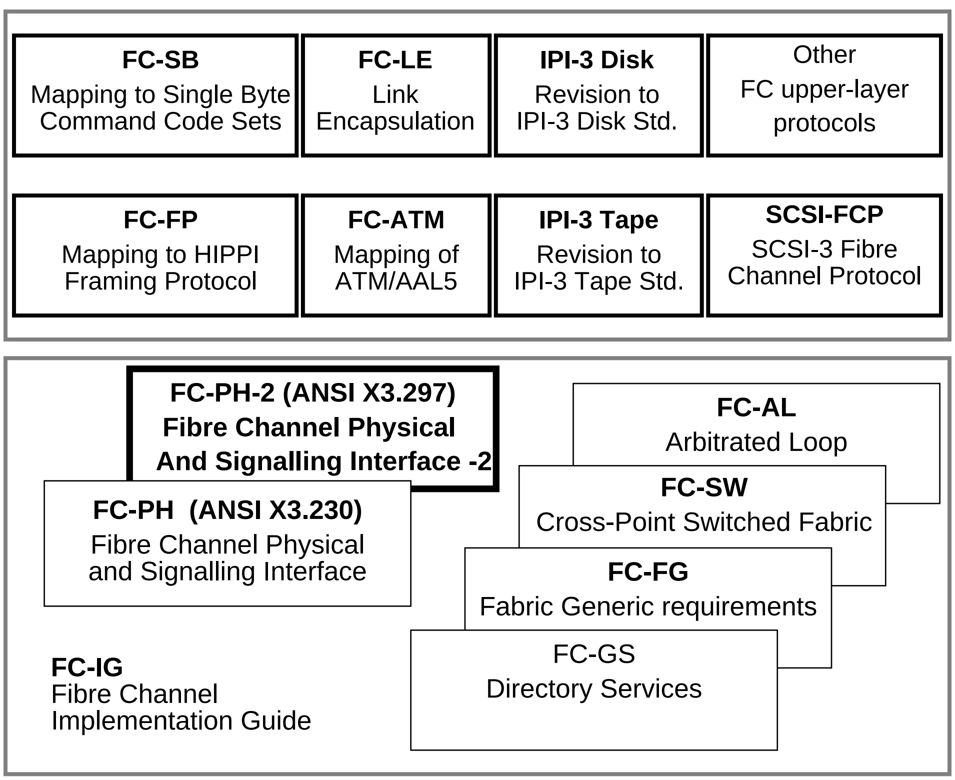
Roger Cummings, Chair
Carl Zeitler, Vice-Chair
KC Chennappan, FC-PH-2 Technical Editor

D. Allan
T. Anderson
K. Annamalai
C. Beck
B. Bellino
C. Brill
K. Chan
K. Chennappan
H. Chin
R. Cook
J. Coomes
R. Cornelius
R. Dahlgren
S. Dean
J. Dedek
F. DeNap
M. Dorsett
B. Edge
S. Erler
E. Freeman
E. Frymoyer
T. Fung
B. Gallagher
C. Grant
M. Griffin
E. Grivna
J. Guedj
D. Hackler
D. Hagerman
T. Harper
N. Harris
V. Haydu
D. Hepner
J. Himes
K. Jackson
E. Jacques
T. Johnson
S. Joiner
R. Kembel
J. Knickerbocker
0. Kornblum
M. Krzych
D. LaFollette
L. Lamers
J. Lear
M. Leib
R. Leibow
K. Malavalli
S. Malladi
C. Masog
B. Masterson
J. Mathis
V. Mattella
K. Mehta
G. Milby
M. Miller
S. Mindemann
A. Miura
M. Montana
J. Mork
J. Morris
J. Murdock
M. O'Donnell
D. Olsen
T. Palkert
M. Parvaresh
A. Patel
R. Pedersen
M. Peterson
G. Porter
R. Prentice
G. Rara
J. Renwick
W. Rickard
E. Rodriguez
R. Ronald
G. Rossmann
P. Rupert
F. Rutherford
P. Scott
M. Shiflett
L. Sloan
K. Smith
R. Snively
D. Somes
J. Sostarich
T. Sprenkle
J. Stai
G. Stephens
B. Støvhase
A. Stone
T. Szostak
R. Taborek
F. Tarverdians
J. Thatcher
L. Thorsbakken
D. Tolmie
J. Toy
H. Truestedt
S. van Doorn
E. Vegdahl
T. Vrankar
R. Wagner
P. Walford
N. Wanamaker
G. Warden
R. Whiteman
J. Williams
K. Witte
J. Young
B. Yunker

Introduction

A set of advanced capabilities to FC-PH are provided in FC-PH-2 to support some sophisticated application requirements.The advanced capabilities include features such as:

- Rotary Group and Hunt Group
- Multicast/Moviecast
- Dedicated Simplex
- Camp-on
- Stacked Connect Request
- Enhancements to ACK protocol, F_CTL, and Timer
- Fractional Bandwidth (Class 4)
- Data Compression
- Higher Data Rates (2 and 4 Giga baud range)
- and others.



Document relationship

These capabilities are accomplished by defining FC-3 Common Services, and enriching and complementing FC-2 Signaling protocol and FC-0 physical media and transceivers defined in FC-PH:

- FC-3 defines a set of services which are common across multiple ports of a node.
- FC-2 enhancements include alias address definition, protocols for Hunt Group, Multicast, Dedicated Simplex, Camp-On, Stacked Connect Request, Fractional Bandwidth allocation management. (Class 4), Data Compression and others.
- FC-0 defines higher data rates in 2 and 4 Gigabaud range.

Figure on page xx shows the relationship of this American National Standard (the highlighted rectangle) with other Fibre Channel documents. ANSI X3.297, FC-PH-2 specifies the enhanced functions added to ANSI X3.230, FC-PH. ANSI X3.289, FC-FG and FC-SW are documents related to Fabric requirements. ANSI X3.272, FC-AL specifies the arbitrated loop topology. ANSI X3.288, FC-GS is a document related to Fibre Channel Services requirements. FC-IG provides some implementation guidance. ANSI X3.271, FC-SB; ANSI X3.254, FC-FP; ANSI X3.287, FC-LE; FC-ATM; IPI-3 Disk; IPI-3 Tape and SCSI-FCP are FC-4 documents.

draft proposed American National Standard

for Information Systems

Fibre Channel —

Physical and Signalling Interface - 2 (FC-PH-2)

1 Scope

ANSI X3.297, FC-PH-2 describes the enhancement to the ANSI X3.230, FC-PH.

This document is an extension to the FC-PH standard and describes a set of advanced capabilities beyond FC-PH to support more advanced and specialized applications.

2 Normative references

The following standards contain provisions which, 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 to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents can be obtained from ANSI: Approved ANSI standards, approved and draft international and regional standards (ISO, IEC, CEN/CENELEC, ITUT), and approved and draft foreign standards (including BSI, JIS, and DIN). For further information, contact ANSI Customer Service Department at 212-642-4900 (phone), 212-302-1286 (fax) or via the World Wide Web at http://www.ansi.org.

2.1 Approved references

ANSI X3.230-1994, Information Technology - Fibre Channel - Physical and Signaling Interface (FC-PH).

2.2 References under development

At the time of publication, the following referenced standards were still under development. For information on the current status of the documents, or regarding availability, contact the relevant standards body or other organization as indicated.

X3 Project 958-D, Information Technology - Fibre Channel - Fabric Requirements (FC-FG) [a]

X3 Project 959-D, Information Technology - Fibre Channel - Switch Topologies and Switch Control (FC-SW) a

---

[a] For information about obtaining copies of this document or for more information on the current status of the document, contact the X3 Secretariat at http://www.x3.org or 202-626-5738.

3 Definitions and conventions

For FC-PH-2, the following definitions, conventions, abbreviations, acronyms, and symbols apply, in addition to those of FC-PH.

3.1 Definitions

3.1.1 Active Virtual Circuit Credit limit: The maximum number of VC_Credits available for a Virtual Circuit in the Active state. It represent the maximum number of VC_Credits held by an N_Port on a given Virtual Circuit when the Circuit is in the active state.

3.1.2 Alias: Alias is a group address recognized by an N_Port if the N_Port has registered as a member of the group, with the Alias Server (see 32.1).

3.1.3 Alias_Token: A 12 byte field to indicate the type of Alias (such as Multicast, Hunt Group) and certain properties associated with the Alias (such as FC-PH TYPE, Node Name for the Common Controlling Entity).

3.1.4 Available BB_Credit: The difference between the number of R_RDYs received and the number of frames sent in a single circuit.

3.1.5 Class 4 bidirectional circuits: A pair of unidirectional Virtual Circuits between two communicating N_Ports.

3.1.6 Class 4 end-to-end credit limit: The maximum amount of end-to-end credit available for a Virtual Circuit. It represents the maximum number of Class 4 end-to-end credit held by an N_Port on a given Virtual Circuit.

3.1.7 Class 4 service: A service that establishes Virtual Circuits to provide fractional bandwidth service between communicating N_Ports. The service multiplexes frames at frame boundaries to or from one or more N_Ports with acknowledgment provided.

3.1.8 Connection-oriented frames: Frames sent in either a Class 1 Dedicated Connection or a Class 4 Circuit.

3.1.9 fractional bandwidth: A portion of the total bandwidth available on a path.

3.1.10 Class 4 Circuit Initiator: The N_Port which initiates the setup of a Class 4 Circuit.

3.1.11 Class 4 Circuit Recipient: The N_Port which accepts a Class 4 Circuit with the Originator N_Port.

3.1.12 Dedicated Simplex: A unidirectional Class 1 connection with ACKs transmitted in Class 2 (see clause 33).

3.1.13 Dedicated Duplex: A synonym for Class 1 Dedicated Connection (see clause 33).

3.1.14 Virtual Circuit (VC): A unidirectional path between two communicating N_Ports that permits Class 4 service to be used. Two Virtual Circuits are required to form a Class 4 Circuit.

3.1.15 Virtual Circuit Credit (VC_Credit): The number of receiver buffers allocated to a Virtual Circuit by an F_Port. It represent the maximum number of frames that an N_Port may transmit without causing a buffer overrun condition at the F_Port receiver.

3.1.16 Virtual Circuit Credit limit: The maximum number of VC_Credits available for a Virtual Circuit. It represent the maximum number of VC_Credits held by an N_Port on a given Virtual Circuit.

3.1.17 Virtual Circuit Identifier (VC_ID): An identifier associated with either the Originator (OVC_ID) or Responder (RVC_ID) for a Virtual Circuit.

3.1.18 virtual path: A fixed route through a Fabric in support of a Virtual Circuit.

3.2 Editorial conventions

Enhancement to FC-PH conventions is specified.

In the various ladder diagrams that show a sequence of events, the vertical axis (i.e., up and down the page) shows time from top to bottom.

The ISO convention of numbering is used i.e., the thousands and higher multiples are separated by a space. A comma is used as the decimal point. A comparison of the American and ISO conventions are shown:

ISO	American
0,6	0.6
1 000	1,000
1 323 462,9	1,323,462.9

Reference to FC-PH in this document implies X3.230 FC-PH document. For example, FC-PH table XX refers to table XX of the X3.230 FC-PH document. FC-PH YY.Z refers to subclause YY.Z of the X3.230 FC-PH document.

3.3 Abbreviations, acronyms and symbols

Abbreviations, acronyms and symbols applicable to this standard are listed. Definitions of several of these items are included in 3.1 The index at the back of the document is an aid to help locate these terms in the body of the document.

3.3.1 Data Rate Abbreviations

Abbreviation	*Data Rate*
133 Mb/s	132,812 5 Mbaud
266 Mb/s	265,625 Mbaud
531 Mb/s	531,25 Mbaud
1 063 Mb/s	1 062, 5 Mbaud
2 125 Mb/s	2 125 Mbaud
4 250 Mb/s	4 250 Mbaud

The exact data rates are used in the tables and the abbreviated form is used in text. Note that 2,125 Gbaud is the preferred ISO method and is used instead of 2 125 Mbaud where it makes sense to do so.

3.3.2 Synonyms

Listed terms are synonyms:

ACC ≡ LS_ACC

3.3.3 Acronyms and other abbreviations

CTI	Circuit Initiator
CTR	Circuit Recipient
COR	Camp-On Request
FDACT	Fabric Deactivate Alias
FACT	Fabric Activate Alias
GAID	Get Alias_ID
HG	Hunt Group
JNA	Join ALias Group
LS_ACC	Link Service Accept
NACT	N_Port Activate Alias
NDACT	N_Port Deactivate Alias
PDISC	Discover N_Port Service Parms
PRLI	Process Login
PRLO	Process Logout
OVC_ID	Originator VC_ID
QoSF	Quality of Service Facilitator
QoSR	Quality of Service Request
RVCS	Read Virtual Circuit status
RVC_ID	Responder VC_ID
SCN	State Change Notification
SCR	Stacked Connect Request
TPLS	Test Process Login Status
VC	Virtual Circuit
VC_Credit	Virtual Circuit Credit
VC_ID	Virtual Circuit Identifier
VC_RDY	Virtual Circuit Ready

3.3.4 Symbols

Unless indicated otherwise, the following symbols have the listed meaning.

λ wavelength

4 Structure and concepts

Enhancements to FC-PH structure and concepts are described. Figure 2 illustrates enhancements to FC-PH figure 2.

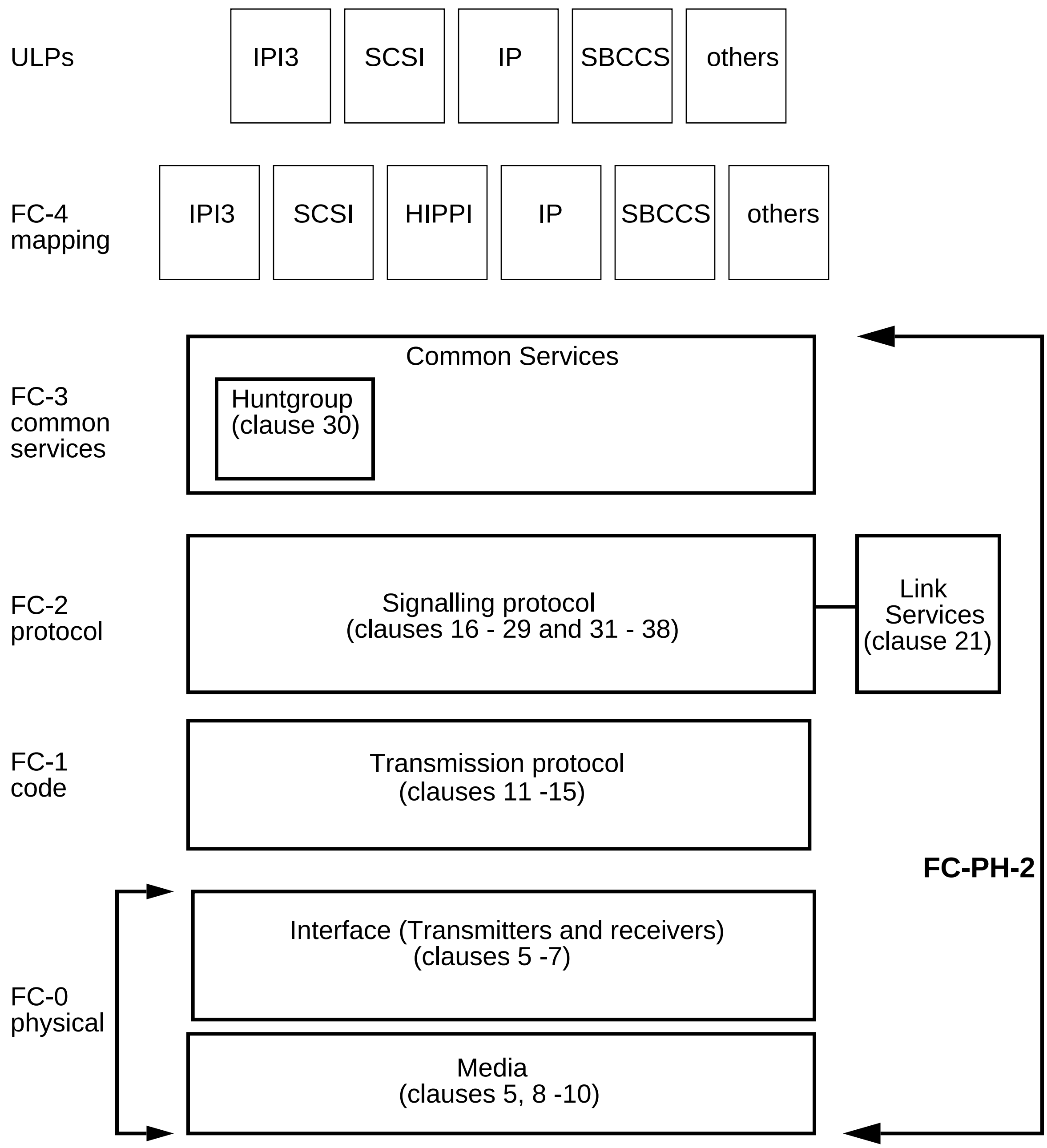


Figure 2 – Fibre Channel Structure

4.6 Communication models

FC-PH-2 defines additional communication models.

4.6.1 Hunt Group

A Hunt Group is an enhancement to the FC-PH structure of N_Ports. In FC-PH N_Ports are addressed using Port specific N_Port Identifiers. The Hunt Group is a collection of N_ Ports which may be addressed by either a group identifier (Hunt Group ID) or Port specific N_Port Identifier. Grouping N_Ports and addressing them as a group makes it possible to increase bandwidth and decrease latency significantly.

Refer to clause 30 for details on Hunt Group.

4.6.2 Dedicated Simplex

Dedicated Simplex approach allows an N_Port to have two separate dedicated simplex connections with two other N_Ports, by using inbound and outbound fibres separately, thereby increasing the bandwidth.

Refer to clause 33 for details on Dedicated Simplex.

4.6.3 Fractional Bandwidth

Class 4 provides the capability to allocate and guarantee bandwidth for multiple N_Ports communicating with a single N_Port. Class 4 also provides a way to support service quality through quality of service parameters which may be specified by an N_Port.

Refer to clause 34 for details on Class 4.

5 FC-0 functional characteristics

The enhancement to FC-PH is specified.

5.1 General characteristics

The FC-2 protocol is defined to operate across connections having a BER detected at the receiving node of 10^{-12}. It shall be the combined responsibility of the component vendors and the system integrator to ensure that this level of service is provided in a given Fibre Channel installation.

The general characteristics specified in FC-PH 5.1 are enhanced as follows:

— Enhanced Serial data streams are supported at data rates of 2 125, and 4 250 MBaud in addition to FC-PH data rates of 133, 266, 531, 1 063 (see 3.3 for exact rates). All rates have clock tolerances of ±100 ppm.

5.7 FC-0 nomenclature

The nomenclature for the technology options are illustrated in figure 21.

100-SM-LL-L

speed
400 400 MB/sec
200 200 MB/sec
100 100 MB/sec
50 50 MB/sec
25 25 MB/sec
12 12,5 MB/sec

media
SM single-mode
M5 multimode (50µm)
M6 multimode (62,5µm)
TV video cable
MI miniature cable
TP twisted pair

transmitter
LL long wave laser
SL short wave laser w/ OFC
SN short wave laser w/o OFC
LE long wave LED
EL electrical

distance
L long distance
I intermediate distance
S short distance

Figure 21 – FC-0 nomenclature

5.8 FC-0 technology options

Optical media signal interface enhancements are included in table 2 based on FC-PH table 2.

Optical cable plant enhancements are included in table 4 based on FC-PH table 4.

Table 2 – Optical Media Signal Interface Overview

400 MB/sec 4,250 Gbaud					**Ref.**
400-SM-LL-I Subclause 6.1 SM 1 300 nm 2m-2km	**400-M5-SN-I** Subclause 6.2.4 MM 780 nm 2m-175m				
200 MB/sec 2,125 Gbaud					
200-SM-LL-I Subclause 6.1 SM 1 300 nm 2m-2km	**200-M5-SN-I** Subclause 6.2.4 MM 780 nm 2m-300m				
100 MB/sec 1,062 Gbaud					**FC-PH-2**
100-M5-SN-I Subclause 6.2.4 MM 780 nm 2m-500m					
25 MB/sec 265,625 Mbaud					
25-M5-LE-I Subclause 6.3 MM 1300 nm 2m-1,5k					

** additional fibre Cable plant, see annex C.

Table 4 – Optical Cable Plant Overview

Single-mode					**Ref.**
400-SM-LL-I Subclause 6.1 SM 1 300 nm 2m-2km	**200-SM-LL-I** Subclause 6.1 SM 1 300 nm 2m-2km				**FC-PH-2**
Multimode (62,5µm)					
400-M6-SN-I** Subclause 6.2.4 MM 780 nm 2m-50m	**200-M6-SN-I**** Subclause 6.2.4 MM 780 nm 2m-90m	**100-M6-SN-I**** Subclause 6.2.4 MM 780 nm 2m-300m			
Multimode (50µm)					
400-M5-SN-I Subclause 6.2.4 MM 780 nm 2m-175m	**200-M5-SN-I** Subclause 6.2.4 MM 780 nm 2m-300m	**100-M5-SN-I** Subclause 6.2.4 MM 780 nm 2m-500m			

** additional fibre cable plant, see annex C.

6 Optical fibre interface specification

Enhancement to FC-PH is specified.

6.1 SM data links

Table 6 includes enhanced data rates of 2 125 and 4 250 MBaud. The optical power coupled into the fibre shall be limited to a maximum value consistent with Class 1 laser safety operations in accordance with IEC 825-1.

Table 6 – FC-0 physical links for single-mode classes

FC-0	Units	400-SM -LL-I	200-SM -LL-I
Subclause		FC-PH-2 6.1	FC-PH-2 6.1
Data rate	MB/sec	400	200
Nominal bit rate	MBaud	4 250	2 125
Tolerance	ppm	±100	±100
Operating range	m	2-2k	2-2k
Fibre core diameter	µm	9	9
Transmitter (S)			
Type		Laser	Laser
λ (1)	nm (min.)	1 270	1 270
	nm (max.)	1 355	1 355
RMS spectral width	nm (max.)	6	6
Launched power, max.	dBm (ave.)	-3	-3
Launched power, min.	dBm (ave.)	-12	-12
Extinction ratio	dB (min.)	9	9
RIN_{12} (max.)	dB/Hz	-116	-116
Eye opening (2)	% (min.)	57	57
Deterministic jitter	% (p-p)	20	20
Receiver (R)			
Received power, min.	dBm (ave.)	-20	-20
Received power, max.	dBm (ave.)	-3	-3
Optical path penalty	dB (max.)	2	2
Return loss of receiver	dB (min.)	12	12

(1) Spectral centre wavelength

(2) @BER$\leq 10^{-12}$

6.2 SW laser data links with OFC

No change from FC-PH.

6.2.4 SW laser data links without OFC

The three columns in table 9B give the link budget, transmitter specifications, and receiver specifications for the short wavelength (SW) laser links operating at the 4 250, 2 125, and 1 063 MBaud rates operating without Open Fibre Control (OFC). The specifications in table 9B are intended to allow compliance to class 1 laser safety.

Proper system performance is ensured by considering the attenuation, reflection, and dispersion characteristics of the optical path and including them as part of the link budget. In addition to these cable plant characteristics, a system power penalty must be included in the link budget. The power penalty includes the effect of eye closure due to transmitter characteristics (finite

Table 9B – FC-0 physical links for shortwave multimode classes without OFC

FC-0	Units	400-M5-SN-I	200-M5-SN-I	100-M5-SN-I
Subclause		6.2.4	6.2.4	6.2.4
Data rate	MB/sec	400	200	100
Nominal bit rate	MBaud	4 250	2 125	1 062,5
Tolerance	ppm	±100	±100	±100
Operating range	m	2-175	2-300	2-500
Fibre core diameter	µm	50	50	50
Transmitter (S)				
Type		Laser	Laser	Laser
λ (1)	nm (min.)	770	770	770
	nm (max.)	860	860	860
Spectral width (max.)	nm RMS	4	4	4
	nm FWHM	NA	NA	NA
Launched power, max.	dBm (ave.)	-5	-5	-5
Launched power, min.(3)	dBm (ave.)	-10	-10	-10
Extinction ratio	dB (min.)	9	9	9
RIN_{12} (max.)	dB/Hz	-120	-118	-116
Eye opening (2)	% (min.)	57	57	57
Deterministic jitter	% (p-p)	20	20	20
Random jitter	% (p-p)	NA	NA	NA
Optical rise/fall time(4)	ns (max.)	0,11	0,22	0,45
Receiver (R)				
Received power, min.(3)	dBm (ave.)	-16	-16	-16
Received power, max.	dBm (ave.)	0	0	0
Return loss (receiver)	dB (min.)	12	12	12
Deterministic jitter	% (p-p)	NA	NA	NA
Random jitter	% (p-p)	NA	NA	NA
Optical rise/fall time	ns (max.)	0,15	0,3	0,6

(1) Spectral centre wavelength

(2) @BER≤10^{-12}

(3) A 6 dB system budget is specified for both 50 um fiber and 62.5 um fiber. Of this 6 dB budget, 4 dB is allocated for cable plant losses and 2 dB is allocated for a system power penalty as described in clause 6.2.4

(4) The optical t_r and t_f for lasers including VCSELs are specified from 20% to 80%, and LED devices from 10% to 90%.

(5) Maximum launched power may exceed -5 dBm as long as launched power does not exceed 0 dBm and complies with applicable laser safety standards.

rise and fall times, random and systematic jitter) and modal noise. The total system link budget is 6 dB (see table 9B), of which 4 dB is allocated for cable plant characteristics and 2 dB for system power penalties.

The attenuation range specification of 0 dB to 4 dB (for each point-to-point link) is defined based on typical premises cabling layouts as described in ANSI/TIA/EIA-568-A and ISO/IEC 11801. The static attenuation in the optical path includes worst case loss values for the fiber media, connectors, splices, attenuators and any other passive optical devices. The attenuation range is 0 to 4 dB for both 50 µm core diameter and 62.5 µm core diameter fiber.

Fiber bandwidth is affected by chromatic dispersion and modal dispersion of the fiber. SW laser based sources are limited by fiber modal dispersion and not chromatic dispersion. The 50 µm core diameter fiber has a minimum modal bandwidth of 500 MHz-km at 850 nm when measured in accordance with ANSI/TIA/EIA-455-51 or -30, the 62.5 µm core diameter fiber has a minimum modal bandwidth of 160 MHz-km at 850 nm when measured with the same procedure. The system implementer may use higher modal bandwidth fiber if available.

Reflection effects on the transmitter are assumed to be small but need to be bounded. A specification of maximum Relative Intensity Noise (RIN) under worst case reflection conditions is included (see table 9B) to ensure that reflections do not impact system performance.

A portion of the system power penalty is reserved for performance degradation due to modal noise. Modal noise can be kept within acceptable limits by controlling the connector loss in the link and by proper transmitter design. A measurement of modal noise is used to ensure the system power penalty budget is satisfied. The system power penalty due to modal noise shall not exceed 1.0 dB for the 50 µm fiber, and not exceed 0.5 dB for the 62.5 µm fiber. This modal noise power penalty assumes that the connector loss is distributed across the link and that the maximum connector loss for any single connector does not exceed 0.6 dB.

6.2.4.1 Optical output interface

The characteristic of the transmitted signal at the optical output interface are given in table 9B. The SW laser described for these links shall have properties that significantly reduce the noise problems associated with using lasers on multimode fiber. In particular, the laser shall be chosen to minimize the effects of modal noise which may occur if the cable plant contains mode selective loss (e.g. poor connections).

Modal noise is worst when the fiber dispersion, between the transmitter and the location of mode selective loss, occurs in an initial length of fiber following the transmitter. A common technique for reducing modal noise is to use a laser with short coherence length. This in turn reduces the initial length of fiber in which modal noise is most problematic. Within this initial length of fiber, extra care shall be taken to avoid mode selective loss. Various oscillation techniques can be used to shorten a laser's coherence length. These oscillation techniques include self-pulsating lasers (which are limited to lower data rates), relaxation oscillations, superimposing an external high frequency oscillating drive signal and the effect from high frequency data modulation (effective at gigabit data rates for some laser structures).

The maximum and minimum of the allowed range of average transmitter power coupled into the fibre are worst-case values to account for manufacturing variances, drift due to temperature variations, aging effects, and operation within the specified minimum value of the extinction ratio.

The minimum value of the extinction shall be the minimum acceptable value of the ratio (in decibels) of the average optical energy in a logic one value to the average optical energy in a logic zero level. The extinction ratio shall be measured under fully modulated conditions in the presence of worst-case reflections.

The transmitter signal must not only meet eye opening requirements, but also overshoot and undershoot limitations. The parameters specifying the (normalized) mask of the transmitter eye diagram are identical to those presented in FC-PH 6.1.1 and shown in figure 22 of FC-PH.

6.2.4.2 Optical input interface

The receiver shall operate at a BER of 10^{-12} over the link's lifetime and temperature range when the input power falls within the range given in table 9B and when driven by a data stream output that fits the specified eye diagram mask through a cable plant as specified in 8.3 or the alternate cable plant in annexes C or D.

Receiver characteristics specified in this document are receiver sensitivity, overload, reflectance, and the allowable power penalty from the combined effects of dispersion, reflections, and jitter.

The minimum and maximum values of the average received power in dBm give the input power range to maintain a BER $\leq 10^{-12}$. These values take into account power penalties caused by the use of a transmitter with a worst-case combination of transmitter spectral, extinction ratio, and pulse shape characteristics specified for each version.

The system power penalty (in decibels) typically accounts for the total degradation along the optical path resulting from reflections, jitter, and the combined effects of dispersion from intersymbol interference, modal noise, mode-partition noise, and laser chirp. However, it is a common practice for multi-mode fibre data links, to include this optical path power penalty into the total link budget specifications of both the power budget (producing amplitude degradation) and the timing budget (producing pulse spreading degradation). Therefore, the SW laser data links specified in table 9B have a total 6 dB system budget, of which 4 dB is allocated for cable plant losses and 2 dB for system power penalties.

A unique power penalty consideration for SW laser data links on multimode fibre is mode selective loss. Refer to FC-PH annex C for design considerations related to mode selective loss.

The minimum acceptable value for receiver sensitivity shall equal the values specified in table 9B. In addition, the receiver shall be designed to accommodate the maximum received power and the optical rise/fall time at the input to the receiver.

6.2.4.3 Safety redundancy

The output from a Safety Class 1 laser product shall not exceed the maximum permissible exposure limits under any conditions of operation, including reasonable single fault conditions per IEC 825-1.

6.2.4.4 Safety documentation and usage restrictions

All laser safety standards and regulations require that the manufacturer of a laser product provide information about a product's laser, safety features, labeling, use, maintenance and service. As part of this documentation, the following usage restriction shall be included with all shortwave (SN) laser transceiver products:

> A type SN laser product not implementing OFC (non-OFC) shall in no case attempt to bring up a link with an OFC compliant product, thereby defeating the safety controls built into the OFC product. Failure to comply with this usage restriction may result in incorrect operation of the link and create points of access that may emit laser radiation above the limit for Safety Class 1 systems established by one or more national or international laser safety standards.

The certification ensures that each of the products will function correctly in the event of a fault in one of the safety control systems.

6.3 LED data links

The single column in table 9C gives the link budget, transmitter specifications, and receiver specifications for the long wavelength (LED) laser links operating at the 265,625 MBaud rate.

Table 9C– FC-0 physical links for longwave multimode classes

FC-0	Units	25-M5-LE-I
Subclause		6.3
Data rate	MB/sec	25
Nominal bit rate	MBaud	265,625
Tolerance	ppm	±100
Operating range	m	2-1,5k
Fibre core diameter	µm	50
Transmitter (S)		
Type		LED
λ (1)	nm (min.)	1 280
	nm (max.)	1 380
Spectral width (max.)	nm RMS	NA
	nm FWHM	fig 6
Launched power, max.	dBm (ave.)	-14
Launched power, min.(3)	dBm (ave.)	-23,5
Extinction ratio	dB (min.)	10
RIN$_{12}$ (max.)	dB/Hz	NA
Eye opening (2)	% (min.)	NA
Deterministic jitter	% (p-p)	16
Random jitter	% (p-p)	9
Optical rise/fall time(4)	ns (max.)	2,0/2,2
Receiver (R)		
Received power, min.(3)	dBm (ave.)	-29
Received power, max.	dBm (ave.)	-14
Return loss (receiver)	dB (min.)	NA
Deterministic jitter	% (p-p)	19
Random jitter	% (p-p)	9
Optical rise/fall time	ns (max.)	2,5

(1) Spectral centre wavelength

(2) @BER≤10^{-12}

(3) The optical t_r and t_f for LED devices are specified from 10% to 90%.

7 Electrical cable interface specification

7.1 Electrical ECL data links

Electrical data links shall have AC coupling at the receiver and may have AC coupling at the transmitter.

8 Optical fibre cable plant specification

8.1 SM cable plant specification

This subclause specifies a single-mode cable plant (see FC-PH 3.1.17 for definition) for the Fibre Channel data rates of 2 125 and 4 250 MBaud.

Table 12 – Single-mode cable plant

FC-0		400-SM-LL-I	200-SM-LL-I
Subclause		6.1	6.1
Operating Range	m	2-2k	2-2k
Cable Plant Dispersion	ps/nm	12	12
Dispersion Related Penalty	dB	1	1
Reflection Related Penalty	dB	1	1
Loss Budget	dB	6	6

8.2 MM 62,5 µm cable plant specification

No change from FC-PH.

8.3 MM 50 µm cable plant specification

This subclause specifies multimode 50 µm cable plant for 780 nm CD laser and the 25-M5-LE-I 1 300nm long wavelength LED.

Table 16 – 50 µm cable plant

FC-0	400-M5-SN-I	200-M5-SN-I	100-M5-SN-I	25-M5-LE-I
Subclause	6.2.4	6.2.4	6.2.4	6.3
Data Rate (M B/s)	400	200	100	25
Operating Range (m)	2-150	2-300	2-500	2-1 500
Loss Budget (dB)	4	4	4	5,5

9 Electrical cable plant specification

No enhancement from FC-PH.

10 Optical interface connector specification

No enhancements from FC-PH.

11 FC-1 8B/10B Transmission Code

11.4 FC-PH-2 Ordered Sets

Tables 24 and 25 provide FC-PH-2 enhancements based on FC-PH tables 24, 25, and 26.

Table 24 – FC-PH Delimiters

Abbr.	Delimiter Function	Beginning RD	Ordered Set				Ref.
Class_4 Specific							Clause 34
SOF(c4)	SOF Activate Class_4	Negative	K28.5	D21.5	D25.0	D25.0	
SOF(i4)	SOF Initiate Class_4	Negative	K28.5	D21.5	D25.2	D25.2	
SOF(n4)	SOF Normal Class_4	Negative	K28.5	D21.5	D25.1	D25.1	
Common EOF Delimiters							
EOF(dt)	EOF Disconnect-Terminate - Class 1 EOF Deactivate-Terminate - Class 4	Negative	K28.5	D21.4	D21.4	D21.4	
		Positive	K28.5	D21.5	D21.4	D21.4	
EOF(dti)	EOF Disconnect-Terminate-Invalid (class 1) EOF Deactivate-Terminate-Invalid (Class 4)	Negative	K28.5	D10.4	D21.4	D21.4	
		Positive	K28.5	D10.5	D21.4	D21.4	
EOF(rt)	EOF Remove-Terminate Class_4	Negative	K28.5	D21.4	D25.4	D25.4	
		Positive	K28.5	D21.5	D25.4	D25.4	
EOF(rti)	EOF Remove-Terminate Invalid Class_4	Negative	K28.5	D10.4	D25.4	D25.4	
		Positive	K28.5	D10.5	D25.4	D25.4	

NOTE - This table contains delimiters added for:

- Class 4 (see clause 34)

No new delimiters were added for:

- Dedicated Simplex (see clause 33)
- Camp-On (see clause 35)
- Stacked Connect Request (see clause 36)
- Data compression (see clause 38)

Table 25 – FC-PH Primitive Signals							
Abbr.	**Primitive Signal**	**Beginning RD**	**Ordered Set**				**Ref.**
Class_4 Specific							
VC_RDY	Primitive: Virtual Circuit Ready	Negative	K28.5	D21.7	VC_ID	VC_ID	Clause 34

NOTE - This table contains Primitive Signal added for:

- Class 4 (see clause 34)

No new Primitive Signals were added for:

- Dedicated Simplex (see clause 33)
- Camp-On (see clause 35)
- Stacked Connect Request (see clause 36)
- Data compression (see clause 38)

12 FC-1 receiver and transmitter description

No enhancements from FC-PH.

13 Loopback mode

No enhancements from FC-PH.

14 Diagnostic mode

No enhancements from FC-PH.

15 Transmitter safety

No enhancements from FC-PH.

16 Ordered Sets

FC-PH-2 enhancements to FC-PH Ordered Sets are summarized in 11.4.

17 Frame formats

No enhancements to FC-PH are specified.

18 Frame_Header

18.1 Introduction

This clause describes the changes that are made in FC-PH-2 to the FC-PH Frame_Header (shown in figure 46 of FC-PH). Enhanced FC-PH-2 Frame_Header shall be subdivided into fields as shown in figure 46.

word \ bits	31 … 24	23 … 16	15 … 8	7 … 0
0	R_CTL	D_ID		
1	CS_CTL	S_ID		
2	TYPE	F_CTL		
3	SEQ_ID	DF-CTL	SEQ_CNT	
4	OX_ID		RX_ID	
5	Parameter			

Figure 46 – FC-PH-2 Frame_Header

18.1.1 Terms

The following new term is used with regard to the Enhanced FC-PH-2 Frame_Header:

– Class Specific Control (CS_CTL) field

18.1.2 Applicability

The definition for the Frame_Header shall apply to all classes of service.

18.2 Class Specific Control Field

Word 1, bits 31-24 of the Frame_Header is defined as the Class Specific Control (CS_CTL) field, it contains management information for the class of service identified by the Start of Frame delimiter. The definition of the CS_CTL field is class dependent.

18.2.1 Class 1

The CS_CTL field is defined in table 33A for Class 1:

– Simplex shall be meaningful only on the connect request frame.

– SCR shall be meaningful only on the connect request frame.

– Camp-On shall be meaningful only on the connect request frame.

– BCR shall be meaningful in all Class 1 frames.

Table 33A– CS_CTL field - Class 1

Bit	Abbr.	Value	Meaning
31	Simplex	0 1	Duplex Simplex
30	SCR	0 1	SCR Not requested SCR requested
29	COR	0 1	COR Not requested COR requested
28	BCR	0 1	Unbuffered Class1 requested Buffered Class 1 requested
27-24			Reserved

18.2.2 Class 2

The CS_CTL field is not used by Class 2 and shall be set to zeros (0000 0000) on all Class 2 frames.

18.2.3 Class 3

The CS_CTL field is not used by Class 3 and shall be set to zeros (0000 0000) on all Class 3 frames.

18.2.4 Class 4

The CS_CTL field is defined in table 33B for Class 4:

Table 33B – CS_CTL field - Class 4

Bit	Abbr.	Meaning
31-24	VC_ID	VIrtual Circuit Identifier

– VC_ID shall be meaningful only on the connect request frame.

18.3 Address identifiers

Additional values specified in FC-PH-2 are shown in table 33.

Table 33– Well-known Address identifiers	
Hex value	**Description**
FFFFF0 to FFFFF7	Specified in FC-PH Table 33
FFFFF8	Alias Server (see 32.2)
FFFFF9	Quality of Service Facilitator - Class 4 (QoSF) (see 34.2.4.1)
FFFFFA to FFFFFE	Specified in FC-PH Table 33
FFFFFF	Broadcast Alias_ID (see 31.5)

18.4 Data Structure (TYPE)

Additional Type codes specified in FC-PH-2 are shown in table 36.

Table 36– Type codes - FC-4 (Device_Data and Link_Data)	
Encoded Value Wd 2, bits 31-24	**Description**
0000 0000 to 0010 1111	Specified in FC-PH Table 36
0011 0000 to 0011 0011	Reserved - Scalable Coherent Interface (SCI)
0011 0100 to 0011 0111	MessageWay
0011 1000 to 0011 1111	Reserved
0100 0000 to 0101 1100	Specified in FC-PH Table 36
0101 1101	Fabric Controller
0101 1110 and 1111 1111	Specified in FC-PH Table 36

18.5 Frame Control (F_CTL)

F_CTL summary is provided in table 37.

Bit 18

In Class 1 the E_C bit shall be set to one in the last Data frame of a Sequence to indicate that the N_Port transmitting E_C is beginning the disconnect procedure. The N_Port transmitting E_C set to one on the last Data frame of a Sequence is requesting the receiving N_Port to transmit an ACK frame terminated by EOFdt if the receiving N_Port has completed all Active Sequences. If the receiving N_Port is not able to transmit EOFdt, E_C set to one requests that the receiving N_Port complete all Active Sequences and not initiate any new Sequences during the current Connection.

The E_C bit is only applicable to Class 1 and 4 Service and is only meaningful on the last Data frame of a Sequence. The E_C bit shall be set to zero on a connect-request frame (SOFc1) in order to avoid ambiguous error scenarios where the ACK (EOFdt) is not properly returned to the Connection Initiator (see FC–PH 28.7.2).

See FC–PH 28.7.2 for a discussion on removing Dedicated Connections (E_C bit).

In Class 4 the E_C bit shall be set to one in the last Data frame of a Sequence to indicate that the N_Port transmitting E_C is beginning the deactivation or removal procedure. The N_Port transmitting E_C = 1 on the last Data frame of a Sequence is requesting the receiving N_Port to transmit an ACK frame terminated by EOFdt, in the case of deactivation or EOFrt in the case of removal if the receiving N_Port has completed all active Sequences. If bit R_C = 1 in addition to E_C = 1, then a Class 4 circuit removal is requested otherwise a deactivation is requested.

If the receiving N_Port is not able to deactivate or remove the Class 4 circuit, E_C = 1 requests that the receiving N_Port complete all active Sequences and not initiate any new Sequences during the current activation cycle (see 34.6.4).

Bits 13-12 - ACK_Form

The ACK_Form bits provide an optional assistance to the Sequence Recipient by translating the ACK capability bits in the N_Port Class Service Login Parameters into an F_CTL field accompanying the frame to be acknowledged. ACK_Form is meaningful on all Class 1 or Class 2 Data Frames of a Sequence and on all connect request frames. ACK_Form is not meaningful on Class 1 or Class 2 Link_Control frames, or any Class 3 frames.

– By resetting ACK_Form=00, the Sequence Initiator is indicating that it does not provide ACK generation assistance.

– By setting ACK_Form=01, the Sequence Initiator is indicating that the ACK_1 shall be used as determined during Login.

– By setting ACK_Form=10, the Sequence Initiator is indicating that the ACK_N shall be used as determined during Login

– BY ACK_Form=11, the Sequence Initiator is indicating that ACK_0 shall be used as determined during Login.

Table 36A– ACK_Form

Word 2, F_CTL Bits 13 and 12	
Encode	**Meaning**
00	No assistance provided
01	ACK_1 required
10	ACK_N required
11	ACK_0 required

Bit 11

Data compression status bit (F_CTL bit 11) is used by the Sequence Initiator to indicate to the Sequence Recipient that the Information Category to which the payload in the frame belongs is compressed. The bit is meaningful in all Data frames of the Information Category to which the Data frames belong. The bit is not meaningful on ACK frames, BSY, or RJT frames.

– By resetting Data compression status bit =0, the Sequence Initiator is indicating that the payload in the frame belonging to the Information Category is not compressed.

– By setting Data compression status bit =1, the Sequence Initiator is indicating that the payload in the frame belonging to the Information Category is compressed.

Table 36B– Data compression status	
Word 2, F_CTL Bit 11	
Encode	**Meaning**
0	Uncompressed frame Payload
1	Compressed frame Payload

Bit 8

For Class 1 bit 8 is meaningful on a connect-request (SOFc1) in Class 1. If bit 8 is set = 0, the Dedicated Connection is bidirectional. If a Dedicated Connection is bidirectional, the Connection Recipient may initiate Sequences immediately after the Dedicated Connection is established. If bit 8 is set = 1, the Dedicated Connection is unidirectional and only the N_Port which transmitted the connect request which establishes the Connection shall transmit Data frames.

For Class 1 other than the connect-request, bit 8 is meaningful on the first and last Data frames of a Sequence. After the connect-request with bit 8 set = 1, the Connection Initiator may reset bit 8 = 0 making the Connection bidirectional for the duration of the Connection on the first or last Data frames of a Sequence (i.e., the bit is not meaningful in other Data frames of the Sequence).

The Connection Recipient may request that a unidirectional Connection be changed to a bidirectional Connection by setting bit 8 = 0, in an ACK frame. Once set to zero in an ACK, all subsequent ACKs shall be transmitted with bit 8 = 0. The Connection Initiator is not required to honor the request to become bidirectional. See FC–PH 28.5.3 for additional clarification.

For Class 4 bit 8 is only meaningful on the last Data frame of a Sequence and only if the E_C bit is 1, see the description for bit 18 and 34.6.4.

Table 37 – F_CTL Enhancements to FC-PH definitions

Control Field	Word 2 Bit	Abbre-viation	Description	Ref.
Exchange/Sequence Control				Specified in FC-PH
Exchange Context	23		0 = Originator of Exchange 1 = Responder of Exchange	
Sequence Context	22		0 = Sequence Initiator 1 = Sequence Recipient	
First_Sequence	21	F_S	0 = Sequence other than first of Exchange 1 = First Sequence of Exchange	
Last_Sequence	20	L_S	0 = Sequence other than last of Exchange 1 = Last Sequence of Exchange	
End_Sequence	19	E_S	0 = Data frame other than last of Sequence 1 = Last Data frame of Sequence	
End_Connection (Class 1) or Deactivate Class 4 circuit	18	E_C	0 = Connective active 1 = End of Connection Pending (Class 1) End of live Class 4 circuit	Clause 34
Reclaimed and Reserved (1)	17			Specified in FC-PH
Sequence Initiative	16		0 = hold Sequence Initiative 1 = Transfer Sequence Initiative	
X_ID reassigned	15		0 = X_ID assignment retained 1 = X_ID reassigned	
Invalidate X_ID	14		0 = X_ID assignment retained 1 = Invalidate X_ID	
ACK_Form	13-12		00 = No Assistance provided 01 = ACK_1 required 10= ACK_N required 11 = ACK_0 required	18.5 and Clause 38
Data compression	11		0 = Uncompressed frame Payload 1 = Compressed frame Payload	
Reserved	10			
Retransmitted Sequence	9		0 = Original Sequence transmission 1 = Sequence retransmission	FC-PH
Unidirectional Transmit (Class 1) Remove Class 4 circuit (Class 4)	8	R_C	0 = Bidirectional transmission (Class 1) 1 = Unidirectional transmission (Class 1 0 = Retain or deactivate circuit (Class 4) 1 = Remove circuit (Class 4)	Clause 34

Note (1) F_CTL bit 17 has been reclaimed from FC-PH and is reserved in FC-PH-2.

Table 37 - F_CTL Enhancements to FC-PH definitions(cont.)				
Control Field	**Word 2 Bit**	**Abbreviation**	**Description**	**Ref.**
Continue Sequence Condition	7-6		Last Data Frame - Sequence Initiator 00 = No information 01 = Sequence to follow-immediately 10 = Sequence to follow-soon 11 = Sequence to follow-delayed	Specified in FC-PH
Abort Sequence Condition	5-4		ACK frame - Sequence Recipient 00 = Continue Sequence 01 = Abort Sequence, Perform ABTS 10 = Stop Sequence 11 = Immediate Sequence retransmission requested Data frame (1st of Exchange) - Sequence Initiator 00 = Abort, Discard multiple Sequences 01 = Abort, Discard a single Sequence 10 =Process policy with infinite buffers 11 = Discard multiple Sequences with immediate retransmission	
Relative Offset Present	3		0 = Parameter field not meaningful 1 = Parameter field - relative Offset	
Exchange reassembly	2		Reserved for Exchange reassembly	
Fill Data Bytes	1-0		End of Data field - bytes of fill 00 = 0 Bytes of fill 01 = 1 Byte of fill (last byte of Data field) 10 = 2 Bytes of fill (last 2 bytes of Data field) 11 = 3 Bytes of fill (last 3 bytes of Data field)	

19 Optional headers

No enhancements to Optional headers defined in FC-PH are specified.

20 Data frames and responses

FC-PH-2 enhancements to Data frames and responses defined in FC-PH are specified.

20.2 Data frames

Table 47 provides a summary of Link_Control, Link_Data, and FC-4 Data frame formats, Basic and Extended Link Service Commands defined in FC-PH (Table 47 of FC-PH) and FC-PH-2.

Table 47 – Frame Formats

Frame Type	Link_Control (FT-0)	Acknowledgment	**ACK**	Specified in FC-PH 20.3
			ACK_0, ACK_1, ACK_N	
		Link_Responses	**Busy**	
			F_BSY, P_BSY	
			Reject	
			F_RJT, P_RJT	
		Link_Command	LCR	
			NTY (see 20.3.4.2), END (see 20.3.4.3)	FC-PH-2
	Data (FT-1)	FC-4 Device_ Data	**FC-4 Device TYPEs**	Specified in FC-PH 18.2
			IP, IPI-3, SCSI, SB,...	
		FC-4 Video_Data	**FC-4 Video Types**	
			Reserved	
		Link_Data	**Basic Link Services (FC-PH 21.2)**	Specified in FC-PH clause 21
			ABTS, BA_ACC, BA_RJT, NOP, RMC	
			Extended Link Services (FC-PH 21.3)	
			ABTX, ACC, ADVC, ECHO, ESTC, ESTS,	
			FLOGI, LOGO, LS_RJT, PLOGI, RCS,	
			RES, RLS, RRQ, RSI, RSS, RTV, TEST	
			Extended Link Service - Proc. (see 21.9 to 21.14)	FC-PH-2
			PRLI, PRLO, SCN, TPLS	
			Extended Link Services - Alias (see 21.16 and 21.17)	
			GAID, FACT, FDACT, NACT, NDACT	
			Extended Link Service - Class 4 (see 21.18)	
			QoSR, RVCS	
			Extended Link Services - FC-AL (see 21.19)	
			PDISC, FDISC, ADISC, TPRLO	

Table 48 indicates allowable delimiters for valid Data frames by Class.

Table 48 – Data frame delimiters

Data frame	Delimiters	Ref.
Class 1, 2, and 3	Specified in FC-PH Table 48	FC-PH 20.2
Class 4	SOFc4, SOFi4, SOFn4, EOFn,EOFt	See clause 34
Dedicated Simplex	same as Class 1 Dedicated Duplex, specified in FC-PH Table 48	See clause 33

20.3 Link_Control

Table 49 provides a summary of Link_Control codes based on FC-PH tables 28 and 49.

<table>
<tr><th colspan="7">Table 49 – Link_Control codes</th></tr>
<tr><th colspan="2">Word 0</th><th rowspan="2">Type</th><th>Word 5</th><th rowspan="2">Description</th><th rowspan="2">Abbr.</th><th rowspan="2">Ref.</th></tr>
<tr><th>Routing (bits 31-28)</th><th>Category (bits 27-24)</th><th>Parameter</th></tr>
<tr><td colspan="7">Link_Control</td></tr>
<tr><td rowspan="14">1100</td><td>0000</td><td rowspan="14">NA</td><td>=1</td><td>Acknowledge - 1</td><td>ACK_1</td><td rowspan="10">Specified in FC-PH 20.3</td></tr>
<tr><td rowspan="2">0001</td><td>≠0</td><td>Acknowledge - N</td><td>ACK_N</td></tr>
<tr><td>=0</td><td>Acknowledge - 0</td><td>ACK_0</td></tr>
<tr><td>0010</td><td rowspan="11">NA</td><td>N_Port Reject</td><td>P_RJT</td></tr>
<tr><td>0011</td><td>Fabric Reject</td><td>F_RJT</td></tr>
<tr><td>0100</td><td>N_Port Busy</td><td>P_BSY</td></tr>
<tr><td>0101</td><td>Fabric Busy (to Data Frames)</td><td>F_BSY</td></tr>
<tr><td>0110</td><td>Fabric Busy (to Link_ CTL Frames)</td><td>F_BSY</td></tr>
<tr><td>0111</td><td>Link Credit Reset</td><td>LCR</td></tr>
<tr><td>1000</td><td>Notify (see 20.3.4.2)</td><td>NTY</td></tr>
<tr><td>1001</td><td>End (see 20.3.4.3)</td><td>END</td><td>FC-PH-2</td></tr>
<tr><td>Others</td><td>Reserved</td><td></td><td></td></tr>
</table>

Delimiters:

Table 50 provides a summary of Link_Control codes based on FC-PH table 50.

Table 50 – Link_Control frame delimiters

Data frame	Delimiters	Ref.
Class 1 and 2	Specified in FC-PH Table 50	FC-PH 20.3
Class 4	SOFn4, EOFn, EOFt , EOFdt, EOFdti, EOFrt, EOFrti	See clause 34
Dedicated Simplex	SOFn2, EOFn, EOFt, EOFdt	See clause 33

20.3.2.2 Acknowledge (ACK)

Except in Dedicated Simplex, ACK shall be in the same Class as that of Data frame. In Dedicated Simplex, ACK shall be in Class 2 while the Data frames are in Class 1 (see clause 33).

20.3.3 Reject (P_RJT, F_RJT)

Table 55 provides the enhanced Reject reason codes based on FC-PH table 55.

Table 55 – P_RJT or F_RJT Reason Codes

Word 0 bits 27-24	Description	By
0000 0001 to 0001 0100	Specified in FC-PH table 55	
0001 0101	Class of Service not supported by the entity at 'FFFFFE' (see 23.3.3).	F
0001 0110 to 0001 1010	Specified in FC-PH table 55	
0001 1011	Invalid VC_ID (Class 4)	F
0001 1100	Invalid CS_CTL field	B
0001 1101	Insufficient Resources for VC (Class 4)	B
0001 1110	Dedicated Simplex not supported	B
0001 1111	Invalid Class of service	B
1111 1111	Specified in FC-PH table 55	
Others	Reserved	

20.3.4.1 Link Credit Reset (LCR) Link_Control command

Class 4 behavior shall not be affected by LCR.

20.3.4.2 Notify (NTY)

The Notify (NTY) Link_Control frame shall be used by the Fabric to inform a destination N_Port that a Camp-On exists on its F_Port. The NTY command is a request to the destination N_Port to initiate the End Connection protocol as defined in FC-PH. The NTY command frame shall be transmitted in Class 2.

The NTY frame shall be sent by the Fabric in Class 2 to the destination N_Port of the connect request frame with the Camp-On CS_CTL bit set (=1) when that destination's F_Port is busy with a previous connection. and the destination N_Port supports intermix. If the destination does not support intermix and Class 2, the Fabric shall not transmit the NTY command to it. The NTY, like other Link_Control frames does not require end-to-end credit and does not participate in end-to-end flow control (FC-PH 20.3.4).

Protocol:

– NTY

– No reply frame

Format: FT_0

Addressing:

The D_ID field of the NTY frame designates the destination N_Port of the connect request frame. The S_ID field of the NTY frame designates the Fabric controller.

Payload:

– The NTY Link_Control frame has no payload.

Class of Service

– Class 2

Reply Sequence:

– None

Code Point:

– Word 0, bits 27-24 = 1000

20.3.4.3 End (END)

The END Link_Control command is used by the Dedicated Simplex to remove a Dedicated Simplex Connection, by the Camp-On to remove an outstanding Camp-On request and by the Class 4 to deactivate or remove the Class 4 circuit.

In the case of Dedicated Simplex, the Connection Initiator or Connection Recipient may issue an END Link_Control command with an EOFdt delimiter to remove the Dedicated Simplex connection (see 33.9.2, 33.10.4, and 33.11.2).

In the case of Camp-On, the Connection Initiator may issue an END Link_Control command with an EOFdt delimiter to remove the outstanding Camp-On request (see 35.4).

In the case of Class 4, the END frame shall indicate that the Class 4 circuit is deactivated or removed. The Class 4 circuit is removed if the End Of Frame delimiter is EOFrt or EOFrti. The Class 4 circuit is deactivated if the End Of Frame delimiter is EOFdt or EOFdti. The END Link_Control frame shall be discarded if the End Of Frame delimiter is EOFn, EOFt, EOFni, EOFa or if the end of frame delimiter is missing. Upon the reception of the END Link_Control frame all active Sequences between the communicating N_Ports, using the Class 4 circuit are abnormally terminated.

Exchange and Sequence recovery shall be performed by the communicating N_Ports, at the discretion of the appropriate Upper Level protocol.

N_Ports shall use the END Link_Control frame to remove or deactivate a Class 4 circuit if the N_Port is unable to determine the state of the Class 4 circuit.

The Fabric shall send the END Link_Control command to remove Class 4 circuits when:

– Fabric re-Login is attempted,

– N_Port or the F_Port enters the Offline state,

– Link failure is detected at the N_Port,

– error recovery for the setup process (see 34.2.4.1), or

– internal Fabric failures of the Class 4 circuit path are detected (i.e., the Fabric is unable to provide the guaranteed quality of service).

The Fabric shall send the END Link_Control command to both N_Ports in the Class 4 circuit, unless prevented by Link failure, Offline, etc.

Protocol:

END
No reply frame

Format: FT_0

Addressing: The D_ID field shall designate the N_Port receiving the END Link_Control frame. In the case of Dedicated Simplex, the S_ID shall indicate the Source N_Port. In Class 4, the S_ID and VC_ID field shall designate the N_Port on which behalf the QoSF is transmitting the END command.

Payload:

– The END Link_Control frame has no payload.

Class of Service

– Class 1, 2 or 4

Reply Sequence:

– None

Code Point:

– Word 0, bits 27-24 = 1001

20.5 ACK generation assistance

If a Sequence Recipient supports multiple ACK forms, an indication about the required ACK form by the Sequence Initiator will be an assistance to the Sequence Recipient in generating it.

20.5.1 N_Port Login

20.5.1.1 Capability Indicator

The ACK generation assistance capability is indicated during N_Port Login in word 0, bit 9 of N_Port Class Service Parameters.

To the Initiator Control Flags (D) specified in FC-PH 23.6.8.3, this additional flag is specified in FC-PH-2 (see 23.6.8.3 and figure 67A).

20.5.1.2 Precedence requirements

The ACK precedence requirements specified in FC-PH 20.3.2 are indicated for easy reference in Table 9.

20.5.2 Terms

ACK_Form refers to F_CTL bits 13 and 12 of Frame_Header Word 2.

20.5.3 Applicability

The ACK precedence determined during Login is applicable to all Class 1 Data frames, Class 2 Data frames, and connect-request frames.

ACK_Form is meaningful on all Class 1 or Class 2 Data frames of a Sequence and on all connect-request frames. ACK_Form is not meaningful on Class 1 or Class 2 Link_Control frames or any Class 3 frames.

Table 55A– N_Port Class Service Parameters - ACK support

Sequence Recipient		Sequence Initiator		ACK form required
Word 1		Word 0		
bit 31	bit 30	bit 11	bit 10	
ACK_N	ACK_0	ACK_N	ACK_0	
0	0	0	0	ACK_1
0	0	0	1	ACK_1
0	0	1	0	ACK_1
0	0	1	1	ACK_1
0	1	0	0	ACK_1
0	1	0	1	ACK_0
0	1	1	0	ACK_1
0	1	1	1	ACK_0
1	0	0	0	ACK_1
1	0	0	1	ACK_1
1	0	1	0	ACK_N
1	0	1	1	ACK_N
1	1	0	0	ACK_1
1	1	0	1	ACK_0
1	1	1	0	ACK_N
1	1	1	1	ACK_0

20.5.4 F_CTL bits

F_CTL Bits 13-12 (ACK_Form field) are set by Sequence Initiator to provide an optional assistance to the Sequence Recipient by indicating in this F_CTL field (see Table 8) its ACK capability determined during N_Port Login.

20.5.5 Login rules

Only ACK_1 shall be used for Classes 1 and 2, during or before the establishment of Login parameters. Additional rules are specified for ACK_Form field usage during these conditions:

a. In Classes 1 and 2, ACK_1 shall be used to acknowledge LOGI (PLOGI or FLOGI) and the ACC to LOGI. This rule shall be followed whether LOGI and the ACC to LOGI flow in the same Class 1 connection or two separate connections.
b. If an additional frame with SOFc1 is used to establish Class 1 connection for LOGI or ACC to LOGI, ACK_1 shall be used to acknowledge all these frames including SOFc1 frame. This rule shall be followed whether LOGI and the ACC to LOGI flow in the same Class 1 connection or two separate connections.
c. If ACK generation assistance is not provided, the ACK_Form field in Class 1 or Class 2 LOGI, ACC frames to LOGI, or any SOFc1 frames used shall be set to 00.
d. If ACK generation assistance is provided, the ACK_Form field in Class 1 or Class 2 LOGI, ACC frames to LOGI, or any SOFc1 frames used shall be set to 01.
e. Once established, the ability or inability to provide ACK generation assistance shall not change until Logout or Relogin occurs.

20.5.6 ACK_Form errors

If a Sequence Recipient receives an ACK_Form value which it does not support, it shall issue a P_RJT with the reason code "protocol error".

21 Link Services

21.1 Introduction

No enhancements to FC-PH.

21.2 Basic Link Service commands

No enhancements to FC-PH. FC-PH table 57 is provided without any changes.

Table 57 – Basic Link Service commands

Word 0		Word 1	Word 5	Description	Abbr.	Ref.
Routing (bits 31-28)	Category (bits 27-24)	Type (bits 31-24)	Parameter			
Basic Link Service						
1000	0000	0000 0000	NA	No Operation	NOP	Specified in FC-PH 21.2
	0001			Abort Sequence	ABTS	
	0010			Remove Connection	RMC	
	0011			Reserved		
	0100			Basic Accept	BA_ACC	
	0101			Basic Reject	BA_RJT	
	Others			Reserved		

21.3 Extended Link Services

Table 61 summarizes FC-PH Extended Link Service Requests and Replies.

Table 61 – Extended Link Service Commands

Word 0 Routing (bits 31-28)	Word 0 Category (bits 27-24)	Type	Payload Word 0 (bits 31-24)	Description	Abbr.	Ref.
Extended Link Service Request						
0010	0010	0000 0001	hex '03'	N_Port Login	PLOGI	Specified in FC-PH 21.4 and Table 61
			hex '04'	F_Port Login	FLOGI	
			hex '05'	Logout	LOGO	
			hex '06'	Abort Exchange	ABTX	
			hex '07'	Read Connection Status	RSC	
			hex '08'	Read Exchange Status Block	RES	
			hex '09'	Read Sequence Status Block	RSS	
			hex '0A'	Request Sequence Initiative	RSI	
			hex '0B'	Establish Streaming	ESTS	
			hex '0C'	Estimate Credit	ESTC	
			hex '0D'	Advise Credit	ADVC	
			hex '0E'	Read Time-out Value	RTV	
			hex '0F'	Read Link Status	RLS	
			hex '10'	Echo	ECHO	
			hex '11'	Test	TEST	
			hex '12'	Reinstate Recovery Qualifier	RRQ	
			hex '20'	Process Login	PRLI	FC-PH-2
			hex '21'	Process Logout	PRLO	
			hex '22'	State Change Notification	SCN	
			hex '23'	Test Process Login State	TPLS	
			hex '24'	Third Party Process Logout	TPRLO	
			hex '30'	Get Alias_ID	GAID	
			hex '31'	Fabric Activate Alias_ID	FACT	
			hex '32'	Fabric Deactivate Alias_ID	FDACT	
			hex '33'	N_Port Activate Alias_ID	NACT	
			hex '34'	N_Port Deactivate Alias_ID	NDACT	
			hex '40'	Quality of Service Request	QoSR	
			hex '41'	Read Virtual Circuit Status	RVCS	
			hex '50'	Discover N_Port Service Parm	PDISC	
			hex '51'	Discover F_Port Service Parm	FDISC	
			hex '52'	Discover Address	ADISC	
			Others	Reserved		
Extended Link Service Reply						
0010	0011	0000 0001	hex '01'	Link Service Reject	LS_RJT	Specified in FCPH 21.4 and Table 61
			hex '02'	Accept	ACC	
			Others	Reserved		

21.4 Extended Link Service Requests

Enhancements are specified in 21.9 to 21.19.

21.5 Extended Link Service Replies

No enhancements to FC-PH.

21.5.2 LS_RJT Reason Codes Summary

LS_RJT reason codes for FC-PH-2 enhancements to Extended Link Service protocols are summarized in table 91.

Table 91 – LS_RJT reason code explanation

Encoded Value (Bits 15-8)	Description	Applicable commands
hex ’00’ to hex ‘2C’	See FC-PH Table 91	See FC-PH Table 91
hex ‘30’	No Alias IDs available for this Alias Type	Get Alias_ID
hex ‘31’	Alias ID cannot be activated (no resources available)	Fabric Activate Alias ID, N_Port Activate Alias ID
hex ‘32’	Alias ID cannot be activated (invalid Alias ID)	Fabric Activate Alias ID, N_Port Activate Alias ID
hex ‘33’	Alias ID cannot be deactivated (doesn’t exist)	Fabric Deactivate Alias ID, N_Port Deactivate Alias ID
hex ‘34’	Alias ID cannot be deactivated (resource problem)	Fabric Deactivate Alias ID, N_Port Deactivate Alias ID
hex ‘35’	Service Parameter conflict	N_Port Activate Alias ID
hex ‘36’	Invalid Alias_Token	Get Alias_ID
hex ‘37’	Unsupported Alias_Token	N_Port Activate Alias ID
hex ‘38’	Alias Group can not be formed	Get Alias ID
hex ‘40’	QoS Parm Error	QoSR
hex ‘41’	VC_ID Not Found	QoSR
hex ‘42’	Insufficient Resources to support a Class4 connection	QoSR
Others	Reserved	

21.6 FC-4 Link Service

No enhancements to FC-PH.

21.7 Basic Link Service summary

No enhancements to FC-PH.

21.9 Process Extended link services

These Extended Link Services are optional. These services are considered to be generic to systems designed to support multiple FC-4's. The usage of Extended Link Services by a particular FC-4 is specified in the corresponding FC-4 document. Presently, these functions are required for FC-SB and SCSI-FCP support.

21.10 PRLI/PRLO command codes

When TYPE indicates Extended Link Service, byte 0 of the first word of the first frame of the Payload (LS_Command code) of the request or reply Sequence shall contain the encoded value shown in table 61A. Subsequent frames, if any, for a request or reply Sequence shall only contain additional Payload in the Payload field (i.e. the LS_Command code is not repeated In each frame).

Information Units associated with the Extended Link Services specified in table 61A are described in table 113.

Table 61A – LS_Command codes

Encoded Value (Bits 31-24)	Description	Abbr.
0010 0000	Process Login	PRLI
0010 0001	Process Logout	PRLO
0010 0010	State Change Notification	SCN
0010 0011	Test Process Login State	TPLS

21.11 Process login

The Process Login (PRLI) Extended Link Service request shall be used to establish the operating environment between a group of related processes at the originating N_Port and a group of related processes at the responding N_Port.

Establishing the operating environment may include the establishment of image pairs and the exchange of Service Parameters. The establishment of image pairs is FC-4 independent and is system structure dependent. The exchange of Service Parameters is FC-4 dependent, and if required by a particular FC-4, shall be specified in the corresponding FC-4 standard.

Table 113 – Extended link service information units

IU Name	Category	Content	F/M/L	SI
PRLI	Unsol Control	Process Login	F	T
PRLO	Unsol Control	Process Logout	F	T
SCN	Unsol Control	State Change Notification	F/L	N/A
TPLS	Unsol Control	Test Process Login State	F	T
Legend: **IU Name** is the Information Unit name **Category** is the Information Category **Content** is the Payload content **F/M/L** is the **F**irst/**M**iddle/**L**ast Sequence of an Exchange **SI** is Sequence Initiative - **H**eld or **T**ransferred				

Process Login is long lived. The number of concurrent Process Logins in effect at an N_Port is a function of the N_Port facilities available. Process Login is separate from N_Port Login. Process Login may be either implicit or explicit.

Implicit Process Login

Implicit Process Login is a method of establishing an operating environment by means other than the explicit use of the PRLI Exchange. Specific methods of Implicit Process Login are not defined.

Explicit Process Login

Explicit Process Login is accomplished by using the PRLI Extended Link Service Sequence within a separate Exchange to establish an operating environment.

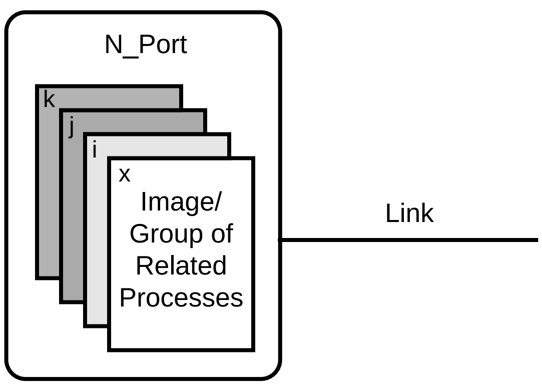


Figure 84 – Images/groups of related processes

A group of related processes is known as an 'image' and is identified by the Process_Associator (see figure 84). The combination of the D_ID, Responder Process_Associator, S_ID and Originator Process_Associator identify the image pair (see figure 85). Either a single group or multiple groups of related processes may exist behind an N_Port. A single group of related processes behind an N_Port may be denoted by either an N_Port ID only or N_Port ID and Process_Associator.

PRLI, if required, is performed after N_Port Login is successful and prior to other IU transfers. Examples of use of the Process Login function may include image initialization, image re-configuration, or when the N_Port receives an indication that the image pair no longer exists. PRLI allows each image behind an N_Port to separately manage its resources.

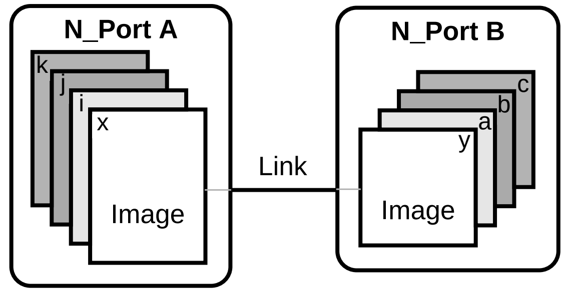


Figure 85 – Image pairs

PRLI may be used to establish an operating environment between any of the following combinations of N_Port facilities:

– two N_Ports;

– one N_Port and one N_Port image;

– two N_Port images.

Multiple image pairs may be established with a single PRLI request and Accept reply Sequence set. Failure to establish a particular image pair does not affect existing image pairs or the ability to establish other image pair.

PRLI may also be used to exchange Service Parameters without establishing image pairs. However, if an image pair is currently established, a subsequent PRLI request targeted to the same N_Port pair shall identify an image pair in order to modify Service Parameter settings for that image pair.

A ULP may choose to establish or modify the operating environment for multiple image pairs with a single PRLI request. This can be accomplished by specifying multiple Service Parameter pages within a single PRLI request. However, a destination N_Port may be unable to execute a PRLI request containing multiple Service Parameter pages. No protocol to determine the capability of a PRLI Responder to execute PRLI requests containing multiple Service Parameter pages is specified. An Accept response containing a non-zero Response code is returned in the case that a PRLI Responder is unable to execute a PRLI request containing multiple Service Parameter pages. (See table 118).

If a PRLI request is received for an established image pair, the existing image pair is unaffected and the PRLI request is processed normally. This allows the exchange of Service Parameters for an FC-4 not specified when the original image pair was established. PRLO shall be used to remove an established image pair.

It shall be the responsibility of the ULPs to ensure that all active operations over an image pair have been properly terminated prior to issuing a PRLI request which replaces Service Parameters. If the replacement of Service Parameters affects any active operations, all active Sequences shall be terminated by invoking Abort Sequence (ABTS) protocol. Following the completion of ABTS protocol, all active Exchanges shall be terminated by invoking either Abort Exchange (ABTX) protocol or Abort Sequence Protocol-Last Sequence (ABTS-LS) protocol. Whether or not the replacement of Service Parameters affects an active operation shall be specified for each Service Parameter by the associated FC-4.

The N_Port originating the PRLI request shall not consider the image pair to be established until it has taken the necessary action to establish the image pair, and has received an Accept reply Sequence indicating that the image pair has been established. The N_Port responding to the PRLI request does not consider the image pair to be established until the necessary action is taken at the N_Port to establish the image pair, and an Accept reply Sequence is sent.

If a link error is detected when a PRLI request is received, the appropriate response, if any, is made, and the image pair is not established. If the PRLI request cannot be accepted for reasons other than a link-error or link-busy condition, an LS_RJT reply Sequence containing the appropriate LS_RJT reason code is sent in response. If an LS_RJT is sent in response to a PRLI request for an image pair that is already established, the existing image pair is unaffected. If an LS_RJT, link-busy, or link-reject response is received to a PRLI request, the PRLI request may be retried until the image pair is established. The number of retries is system dependent. In the case of LS_RJT, whether or not the PRLI is retried depends on the LS_RJT reason code.

In the event that there is an error in the response to establish an image pair, the originating N_Port cannot assume that the requested action has or has not taken place. If no valid response is received by the N_Port to the PRLI request, the N_Port can retry the request. The number of retries is system dependent.

Protocol:

Process Login request Sequence
Accept reply Sequence

Format: FT-1

Addressing: The S_ID field designates the N_Port requesting Process Login. The D_ID field designates the destination N_Port of the Process Login.

Payload: The format of the Payload is shown in table 114.

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Service Parameter page. The rightmost two bits shall be zeros.

Table 114 – PRLI payload

Item	Size -Bytes
x‘20’ = PRLI Command code	1
x‘10’ = Page Length	1
Payload Length	2
Service Parameter pages	16-max

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the PRLI Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

Service parameter page(s): Words 1:max of the PRLI Payload contain one or more Service Parameter pages. Each Service Parameter page contains Service Parameters for a single image pair and is associated with either a single FC-4 TYPE or is common to all FC-4 TYPE codes for the specified image pair.

The format of PRLI Service Parameter pages is described in table 115.

Table 115 – PRLI service parameter page format

Item	Word	Bit
Service Parameter page	0-3	31-0
TYPE Code or Common Service Parameters[1]	0	31-24
TYPE Code Extension	0	23-16
Originator Process_ Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Establish Image Pair	0	13
Reserved	0	12-0
Originator Process_ Associator	1	31-0
Responder Process_ Associator	2	31-0
Service Parameters	3	31-0
NOTE – If byte 0 of the first word of a Service Parameter page is set to the value x'00', the Service Parameter page is common to all FC-4 TYPEs at that image pair.		

- **Word 0, Bits 31-24 - TYPE code or common service parameters**

Identifies the protocol associated with this Service Parameter page. If byte 0 of the first word of a Service Parameter page contains the value x'00', the Service Parameter page contains Service Parameters common to all FC-4 TYPEs at that image pair or N_Port pair. If byte 0 of the first word of a Service Parameter page contains the value other than x'00', the Service Parameter page contains Service Parameters for the FC-4 TYPE indicated as specified by ANSI X3.230, FC-PH.

- **Word 0, Bits 23-16 - TYPE code extension**

Reserved for future use.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 13 - Establish Image Pair**

0 = Exchange Service Parameters only

1 = Establish image pair and exchange Service Parameters

- **Word 3, Bits 31-0 - Service parameters**

Common Service Parameters are specified below. FC-4 TYPE code dependent Service Parameters shall be specified in the corresponding FC-4 standard.

NOTE – No Common Service Parameters are currently specified.

Reply link service sequence:

Link Service Reject (LS_RJT)
  signifies rejection of the PRLI request
Accept (ACC)
  signifies successful completion of the PRLI request
  - Accept Payload
    The format of the Accept Payload is shown in table 116.

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Service Parameter Response page. The rightmost two bits shall be zeros.

Table 116 – PRLI accept payload

Item	Size -Bytes
x'02' = ACC Command code	1
x'10' = Page Length	1
Payload Length	2
Service Parameter Response pages	16-max

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the PRLI Accept Payload. The right-most two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

Service parameter response page(s): Words 1:max of the PRLI Accept Payload contain one or more Service Parameter Response pages. Each Service Parameter Response page contains Service Parameter responses for a single image pair or N_Port pair and is associated with a single FC-4 TYPE or common to all FC-4 TYPEs at that image pair or N_Port pair.

The number of Service Parameter pages and the Payload Length field of the PRLI Accept reply Sequence shall match the number of Service Parameter pages and the Payload Length field of the PRLI request Sequence.

The format of PRLI Accept Service Parameter Response pages is described in table 117.

- **Word 0, Bits 31-24 - TYPE code or common service parameters**

Identifies the protocol associated with this Service Parameter Response page. If byte 0 of the first word of a Service Parameter Response page contains the value x‘00’, the Service Parameter page contains Service Parameters common to all FC-4 TYPEs at that image pair or N_Port pair. If byte 0 of the first word of a Service Parameter Response page contains a value other than x‘00’, the Service Parameter page contains Service Parameters for the FC-4 TYPE indicated as specified by ANSI X3.230, FC-PH.

- **Word 0, Bits 23-16 - TYPE code extension**

Reserved for future use.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

0 = not meaningful

1 = meaningful

Table 117 – PRLI accept service parameter response page format

Item	Word	Bit
Service Parameter Response page	0-3	31-0
TYPE Code or Common Service Parameters[1]	0	31-24
TYPE Code Extension	0	23-16
Originator Process_Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Image Pair Established	0	13
Reserved	0	12
Response Code (see table 118)	0	11-8
Reserved	0	7-0
Originator Process_Associator	1	31-0
Responder Process_ Associator	2	31-0
Service Parameter Response	3	31-0

NOTE – If byte 0 of the first word of a Service Parameter page is set to the value hex’00’, the Service Parameter page is common to all FC-4 TYPEs at that image pair.

- **Word 0, Bit 13 - Image Pair Established**

Image Pair Established is valid only if bit 13 was set to one on the corresponding Service Parameter page of the PRLI request.

0 = Image pair not established, see response code for additional information

1 = Image pair established

- **Word 0, Bits 11-8 - Response code**

The Response code field contains an encoded binary value indicating the result of the PRLI request. The meanings of the encoded Response code values are shown in table 118.

Table 118 – PRLI accept response code

Encoded Value Wd 0, Bits 11-8	Description
0000	Reserved
0001	Request executed
0010	The target image has no resources available for establishing image pairs between the specified source and destination N_Ports. The PRLI request may be retried.
0011	Initialization is not complete for the target image. The PRLI request may be retried.
0100	The target image corresponding to the Responder Process_ Associator specified in the PRLI request and PRLI Accept response does not exist. The PRLI request shall not be retried.
0101	The target image has a predefined configuration which precludes establishing this image pair. The PRLI request shall not be retried.
0110	Request executed conditionally. Some Service Parameters were not able to be set to their requested state. See the Service Parameter Response field for further details.
0111	The destination N_Port is unable to process multiple page PRLI requests. The PRLI request may be retried as a single page request.
1000:1111	Reserved

- **Word 3, Bits 31-0 - Service parameter response**

Provides feedback to the Originator as to the resultant state of the Service Parameters as returned by the Responder.

21.11.1 PRLI/PRLO Relationships

Process_Associator (PA) images can exist in the following relationships. Any of these relationships can be established as a default condition using mechanisms not specified by the standards.

21.11.1.1 PA not supported

If PAs are not supported by one or both of the Originator and Responder, no valid PA is specified in any frame of any exchange.

PRLI and PRLO commands may have no more than a single page for the Common Service Parameters and an additional single page for each additional TYPE supported by the Originator or Responder. Each page indicates that both the Originator PA and Responder PA are invalid.

The PRLI/PRLO can be thought of as enabling and disabling the specified services for the entire N_Port.

21.11.1.2 PA required by originator, supported by responder

If PAs are required by the Originator, the Originator is expected to communicate only when a valid PA is included in the initial Association_Header. The Responder provides a final Association_Header and provides other Association_Headers as required by ANSI X3.230, FC-PH. The PA establishes a routing to a particular Originator process (or group of related processes). A common set of processes is identified by setting the PA of one N_Port of a node to the same value as that of the PA of another N_Port of the same node. ANSI X3.xxx, FC-EP, may define Hunt and Stripe Groups using N_Port aliases. If so, then a process defined by a PA for the alias of the group is considered as the same process for all N_Ports of the group. The PA may additionally be used by the FC-4 in an FC-4 dependent manner.

The PRLI request pages have valid Originator PA values and invalid Responder PA values. This PRLI informs the Responder of the Originator requirements and capabilities for each TYPE of FC-4.

The PRLI and PRLO can be thought of as enabling and disabling the access of each Originator Process to the Responder N_Port. Communication with image pairs that have not been established is not allowed, even if both PAs exist and have the proper requirements and capabilities.

21.11.1.3 PA required by responder, supported by originator

If PAs are required by the Responder, the Responder is expected to communicate only when a valid Responder PA is included in the initial Association_Header. The Responder provides a final Association_Header and provides other Association_Headers as required by ANSI X3.230, FC-PH. The PA establishes a routing to a particular Responder process. A common set of processes is identified by setting the PA of one N_Port of a node to the same value as that of the PA of another N_Port of the same node. ANSI X3.xxx, FC-EP, may define Hunt and Stripe Groups using N_Port aliases. If so, then a process defined by a PA for the alias of the group is considered as the same process for all N_Ports of the group. The PA may additionally be used by the FC-4 in an FC-4 dependent manner.

The PRLI request pages have valid Responder PA values and invalid Originator PA values. The Responder PA values may be obtained through an informative PRLI operation or by other methods not specified.

The PRLI and PRLO can be thought of as enabling and disabling the access of each Responder Process to the Originating N_Port. Communication with image pairs that have not been established is not allowed, even if both PAs exist and have the proper requirements and capabilities.

21.11.1.4 PA required by originator and responder

Communication can only take place between an Originator and a Responder when valid PAs are specified. ANSI X3.xxx, FC-EP, may define hunt groups and striping groups using N_Port aliases. If so, then a process (or group of processes) defined by a PA for the alias of the group is considered as the same process for all N_Ports of the group. The PA may additionally be used by the FC-4 in an FC-4 dependent manner.

The PRLI request pages must have valid Originator and Responder PA values defined and must have the Establish Image Pair bit set to one in order to create image pairs in an binding manner. The Responder PA values may be obtained through an informative PRLI operation or by other methods not specified.

PRLI and PRLO pages with complete Originator PA and Responder PA information enable or disable communication between the specified image pair. PRLI pages with incomplete Originator PA and Responder PA information are invalid. Communication with image pairs that have not been established is not allowed, even if both PAs exist and have the proper requirements and capabilities.

21.11.2 Process login, mode of operation

PRLI has two modes of operation.

Informative mode

Service Parameter information is exchanged enabling subsequent negotiation for image pair establishment.

Binding mode

Information is exchanged that explicitly establishes a relationship between processes in the communicating N_Ports. The relationship does not allow any communication types or paths other than those established by the PRLI.

The use of a Binding PRLI page requires that the Originator have precise and detailed knowledge of the PAs and capabilities available in the Responder. That information may be obtained from Directory Services, implicitly from configuration information obtained outside the scope of FC, or by performing an Informative PRLI.

Binding or Informative mode is determined by the setting of the Establish Image Pair bit in the PRLI request page.

The Service Parameters included in a page may be either requirements or capabilities. Capabilities indicate those FC-4 properties that describe the role and state of the node in the FC-4. Such capabilities include channel or device for ANSI X3.271, FC-SB, initiator or target for ANSI X3.xxx, FCP, and similar values. Requirements indicate those FC-4 properties that must be agreed upon by both nodes for operation with an FC-4. Requirements include values such as the parameters controlling the ANSI X3.xxx, FCP, IUs that are used in communication.

21.11.3 Process login, protocol

21.11.3.1 PA required by originator and responder

For each PRLI request page the Originator and Responder PA validity bits are set to one and valid Originator and Responder PAs are specified. Service Parameter requirements are set to those that correspond to the combined understanding expected to be agreed to by the Responder, given the knowledge that the Originator PA has of the Responder PA. Service Parameter capabilities are those of the Originator only. The specification of Service Parameters as either requirements or capabilities is specified in the corresponding FC-4 standard.

For each ACC response page the Originator and Responder PA validity bits are set to one and valid Originator and Responder PAs are specified. Service Parameter requirements are set to those that are agreed to by the Responder. Service Parameter capabilities are those of the Responder only. Error responses are possible.

Each page identifies a Binding mode PRLI operation between one image at the Originator N_Port and one image at the Responder N_Port.

21.11.3.2 PA required by originator, supported by responder

For each PRLI request page the Originator PA validity bit is set to one and the Responder PA validity bit is set to zero. A valid Originator PA is specified. Service Parameter requirements are set to those that correspond to the combined understanding expected to be agreed to by the Responder, given the knowledge that the Originator PA has of the Responder PA. Service Parameter capabilities are those of the Originator only. The specification of Service Parameters as either requirements or capabilities is specified in the corresponding FC-4 standard.

For each ACC response page the Originator PA validity bit is set to one and the Responder PA validity bit is set to zero. The Originator PA is returned. Service Parameter requirements are set to those that are agreed to by the Responder. Service Parameter capabilities are those of the Responder only. Error responses are possible.

Each page identifies a Binding mode PRLI operation between one image at the Originator N_Port and the Responder N_Port.

21.11.3.3 PA supported by originator, required by responder

For each PRLI request page the Originator PA validity bit is set to zero and the Responder PA validity bit is set to one. A valid Responder PA is specified. Service Parameter requirements are set to those that correspond to the combined understanding expected to be agreed to by the Responder, given the knowledge that the Originator PA has of the Responder PA. Service Parameter capabilities are those of the Originator only. The specification of Service Parameters as either requirements or capabilities is specified in the corresponding FC-4 standard.

For each ACC response page the Originator PA validity bit is set to zero and the Responder PA validity bit is set to one. The Responder PA is returned. Service Parameter requirements are set to those that are agreed to by the Responder. Service Parameter capabilities are those of the Responder only. Error responses are possible.

Each page identifies a Binding mode PRLI operation between the Originator N_Port and one image at the Responder N_Port.

21.12 Process logout

The Process Logout (PRLO) Extended Link Service request shall be used to request invalidation of the operating environment between an image at the initiating N_Port and an image at the recipient N_Port. PRLO frees resources committed by a previous PRLI function. ULP behavior following successful execution of the PRLO function is specified in the corresponding FC-4 standard.

Examples of PRLO usage include image re-configuration and TYPE-specific reset of Process Login Service Parameters.

One or more image pairs may be removed with a single PRLO request and Accept reply Sequence set. Other image pairs associated with the same or different N_Ports shall be unaffected. After an image pair is removed, IUs may not be sent or received using that image pair.

One or more TYPE-specific Service Parameter settings may be reset with a single PRLO request and Accept reply Sequence set. Other TYPE-specific Service Parameter settings associated with the same or different image pairs or N_Ports shall be unaffected. After TYPE-specific Service Parameter settings are reset, IUs may not be sent or received for the specified FC-4 TYPE using that image pair and N_Port combination specified in the PRLO request.

A ULP may choose to remove multiple image pairs or reset multiple TYPE-specific Service Parameter settings with a single PRLO request. This can be accomplished by specifying multiple Logout Parameter pages within a single PRLO request. However, a destination N_Port may be unable to execute a PRLO request containing multiple Logout Parameter pages. No protocol to determine the capability of a PRLO Responder to execute PRLO requests containing multiple Logout Parameter pages is specified. An Accept response containing a non-zero Response code is returned in the case that a PRLO Responder is unable to execute a PRLO request containing multiple Logout Parameter pages.

If a PRLO request is received for an image pair or FC-4 TYPE that does not exist, the request is accepted, provided that no link errors are detected, and the Accept response is sent.

If a link error is detected when a PRLO request is received, the Sequence is discarded, and the appropriate response, if any, for the link error recognized is sent. The N_Port originating the PRLO request shall not consider an image pair to be removed until it receives an error-free Accept reply Sequence. The N_Port responding to the PRLO request shall not consider an image pair to be removed or TYPE-specific Service Parameter settings to be reset, as appropriate, until the Accept reply Sequence is completely sent without error. An N_Port that receives a link-busy or link-reject reply in response to a PRLO request may retry the PRLO request. The number of retries is system dependent.

Unless the requesting N_Port receives a valid response to a PRLO request, that N_Port cannot assume that the requested action has or has not taken place. If an invalid response is received by an N_Port to the PRLO request, the N_Port can retry the request. The number of retries is system dependent.

Protocol:

Process Logout request Sequence
Accept reply Sequence

Format: FT-1

Addressing: The S_ID field designates the N_Port requesting Process Logout. The D_ID field designates the destination N_Port of the Process Logout.

Payload: The format of the Payload is shown in table 119.

Table 119 – PRLO payload

Item	Size -Bytes
x‘21’ = PRLO Command code	1
x‘10’ = Page Length	1
Payload Length	2
Logout Parameter pages	16-max

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Logout Parameter page. The rightmost two bits shall be zeros.

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the PRLO Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

Logout parameter page(s): Words 1:max of the PRLO Payload contain one or more Logout Parameter pages. Each Logout Parameter page contains logout parameters for a single image pair and is associated with either a single FC-4 TYPE or is common to all FC-4 TYPE codes for the specified image pair.

The format of PRLO Logout Parameter pages is described in table 120.

- **Word 0, Bits 31-24 - TYPE code or common logout parameters**

  Identifies the protocol associated with this Logout Parameter page. If byte 0 of the first word of a Logout Parameter page contains the value x‘00’, the Logout Parameter page contains Logout Parameters common to all FC-4 TYPEs at that

Table 120 – PRLO logout parameter page format

Item	Word	Bit
Logout Parameter page	0-3	31-0
TYPE Code or Common Logout Parameters[1]	0	31-24
TYPE Code Extension	0	23-16
Originator Process_ Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Reserved	0	13-0
Originator Process_ Associator	1	31-0
Responder Process_ Associator	2	31-0
Reserved	3	31-0
NOTE – If byte 0 of the first word of a Logout Parameter page is set to the value x'00', the Logout Parameter page is common to all FC-4 TYPEs at that image pair.		

image pair or N_Port pair. If byte 0 of the first word of a Logout Parameter page contains the value other than x'00', the Logout Parameter page contains Logout Parameters for the FC-4 TYPE indicated as specified by ANSI X3.230, FC-PH.

- **Word 0, Bits 23-16 - TYPE code extension**

Reserved for future use.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

0 = not meaningful

1 = meaningful

Reply link service sequence:

Link Service Reject (LS_RJT)

signifies rejection of the PRLO request

Accept (ACC)

signifies successful completion of the PRLO request

- Accept Payload

The format of the Accept Payload is shown in table 121.

Table 121 – PRLO accept payload

Item	Size -Bytes
x'02' = ACC Command code	1
x'10' = Page Length	1
Payload Length	2
Logout Parameter Response pages	16-max

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Logout Parameter Response page. The rightmost two bits shall be zeros.

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the PRLO Accept Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

Logout parameter response page(s): Words 1:max of the PRLO Accept Payload contain one or more Logout Parameter Response pages. Each Logout Parameter Response page contains a logout parameter response for a single image pair and is associated with a single FC-4 TYPE or common to all FC-4 TYPEs at that image pair or N_Port pair.

The number of Logout Parameter pages and the Payload Length field of the PRLO Accept reply Sequence shall match the number of Logout Parameter pages and the Payload Length field of the PRLO request Sequence.

The format of PRLO Accept Logout Parameter Response pages is described in table 122.

Table 122 – PRLO accept logout parameter response page format

Item	Word	Bit
Logout Parameter page	0-3	31-0
TYPE Code or Common Service Parameters[1]	0	31-24
TYPE Code Extension	0	23-16
Originator Process_ Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Reserved	0	13-12
Response Code (see table 123)	0	11-8
Reserved	0	7-0
Originator Process_ Associator	1	31-0
Responder Process_ Associator	2	31-0
Reserved	3	31-0
NOTE – If byte 0 of the first word of a Logout Parameter page is set to the value x'00', the Logout Parameter page is common to all FC-4 TYPEs at that image pair.		

- **Word 0, Bits 31-24 - TYPE code or common logout parameters**

Identifies the protocol associated with this Logout Parameter Response page. If byte 0 of the first word of a Logout Parameter Response page contains the value x'00', the Logout Parameter page contains Logout Parameters common to all FC-4 TYPEs at that image pair or N_Port pair. If byte 0 of the first word of a Logout Parameter Response page contains a value other than x'00', the Logout Parameter page contains Logout Parameters for the FC-4 TYPE indicated as specified by ANSI X3.230, FC-PH.

- **Word 0, Bits 23-16 - TYPE code extension**

Reserved for future use.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bits 11-8 - Response code**

The Response code field contains an encoded binary value indicating the result of the PRLO request and the status of the image pair. The meanings of the encoded Response code values are shown in table 123.

Table 123 – PRLO accept response code

Encoded Value Wd 0, Bits 11-8	Description
0000	Reserved
0001	Request executed
0010:0011	Reserved
0100	The target image corresponding to the Responder Process_ Associator specified in the PRLO request and PRLO Accept response does not exist. The PRLO request shall not be retried.
0101:0110	Reserved
0111	The destination N_Port is unable to process multiple page PRLO requests. The PRLO request may be retried as a single page request.
1000:1111	Reserved

21.12.1 PRLO Operation

PRLO operates by referencing particular PA image pairs. Pages can be mixed in any combination in a PRLO request. The PRLO Accept is required to present a response page for every request page. Pages may be individually marked as being in error.

A PRLO page identifies a particular image pair to logout by marking either or both the Originator Process Associator and the Responder Process Associator. Only that image pair is logged out. No further communication for the affected FC-4(s) is possible between these two images. It is the responsibility of the ULPs to ensure that all active operations over an image pair have been orderly and properly terminated prior to issuing a PRLO request. Following PRLO execution, all active Sequences are terminated by invoking Abort Sequence (ABTS) protocol. Following the completion of ABTS protocol, all active Exchanges shall be terminated by invoking either Abort Exchange (ABTX) protocol or Abort Sequence Protocol-Last Sequence (ABTS-LS) protocol. On-going operations and states for other image pairs are not affected.

A PRLO page identifies a particular FC-4 TYPE to logout via specification of the FC-4 TYPE. Only service parameters for the specified FC-4 TYPE(s) are reset. No further communication for the affected FC-4(s) is possible. It is the responsibility of the ULPs to ensure that all active operations for the affected FC-4(s) have been orderly and properly terminated prior to issuing a PRLO request. Following PRLO execution, all active Sequences are terminated by invoking Abort Sequence (ABTS) protocol. Following the completion of ABTS protocol, all active Exchanges shall be terminated by invoking either Abort Exchange (ABTX) protocol or Abort Sequence Protocol-Last Sequence (ABTS-LS) protocol. On-going operations and states for other FC-4 TYPEs are not affected.

ULP attempts to communicate over an image pair that has not been established or has been abnormally terminated shall be acknowledged in the normal manner. The Originator may then perform a PRLO operation for the affected image pair in order to properly terminate the operating environment at both the Originator and Responder.

21.13 State change notification

A State Change Notification (SCN) Extended Link Service request shall be sent to selected N_Ports when an event occurs which may have affected the current N_Port Login or Process Login (i.e. image pair) states. The term Login state is used here to refer to either or both the N_Port Login state or the N_Port Process Login state. There is no reply to the SCN request Sequence.

The main purpose of SCN is communicating the release of shared resources. If a node fails, timely notification of dependent nodes is important. SCN provides an early warning which, in turn, allows the release of resources dedicated to the failed node. Release of serially-reusable node resources of the failed N_Port or N_Port process may avoid timeouts and deadlocks in the sharing nodes. Early failure data capture, problem notification and determination can begin soon after the point of failure. Release of shared resources to other nodes also has benefits in non-error situations. Accounting and load balancing are two system dependent system management functions which benefit from real-time reporting of Login state changes.

The direction of SCN requests is from the Fabric to its attached nodes. The ultimate notification of an SCN request is from an F_Port to an N_Port. Since the initiative to send an SCN request may have occurred at a source distant from the F_Port directly attached to the N_Port requiring notification (e.g. another N_Port, an intermediate Fabric element, a management server, etc.) SCN protocol shall be described in terms of SCN issued by an F_Port or N_Port.

21.13.1 SCN issued by an F_Port

The Login state changes which cause SCN to be sent from an F_Port are as follows:

– all implicit F_Port Logouts, including loss of signal, NOS, and OLS;

– SCN's sent to the Fabric by an N_Port (a self-generated SCN from an N_Port);

– the Fabric's capability of establishing a communication path between a pair of N_Ports is altered.

When an N_Port changes Login state and the state change is recognized by the Fabric, the

Fabric then issues SCN to selected N_Ports, specifying the N_Port ID of the N_Port which changed state. Any N_Port which was previously logged in to the state-changed N_Port may then test the state of the Login via TPLS, to verify Process Login state, or other Link Services such as NOP, to verify Login state, prior to performing Login state recovery. The Affected N_Port ID is specified in the Payload of the SCN request.

If the D_ID field of the SCN request specifies the well-known address identifier for the Fabric Controller, x'FFFFFD', the Fabric is to report SCN to other N_Ports. The Fabric maintains the ID's of the N_Ports with changed state. For each of these the Fabric also maintains a list of N_Ports requiring notification. For each N_Port requiring notification, the Fabric ignores subsequent N_Port Login state changes until the SCN is sent and a valid response is received. This is to limit the volume of SCN traffic.

The Fabric reports the state change by issuing SCN's to selected N_Ports, specifying the N_Port ID's of the affected N_Ports.

21.13.2 SCN issued by an N_Port

The Login state changes which cause SCN to be sent by an N_Port are as follows:

- all implicit N_Port Logouts. In the case of a node that has two or more N_Ports, if the failure of one of the N_Ports is undetected by the Fabric, then SCN is sent via a surviving N_Port.

An N_Port may optionally report a state change that is caused by an event which affects existing image pairs or affects the ability to accept new image pairs. Individual images behind an N_Port are capable of generating the SCN request. The N_Port reports the state change by sending an SCN request to the Fabric Controller or, optionally, to the destination N_Port directly. However, an N_Port shall not issue an SCN request if it has already signaled a condition which would cause the Fabric, if present, to issue an SCN request on behalf of that N_Port.

If the first SCN request has not been sent by the N_Port, subsequent state changes in that N_Port do not create initiative to send another SCN request. Once the first SCN request has been sent, a subsequent state change creates a new initiative. If the N_Port is sending SCN requests to destinations other than the Fabric Controller, this rule applies separately to each destination.

21.13.3 SCN initiative

Initiative to send an SCN request is discharged when one of the following takes place:

a) An ACK frame is received in response to the SCN request (Class 1 or 2);

b) A condition is recognized for which retry of an SCN request, if previously sent, is not performed;

c) For a condition that does not require retry to be performed indefinitely, the system-dependent number of retries is exhausted without success while attempting to send an SCN request.

21.13.4 Interest registration

The Fabric shall only issue SCN to N_Ports which have registered to be notified of Login state changes for all other N_Ports. One method is to register at the time an N_Port queries the Name Server to obtain the N_Port ID for the N_Port that changed Login state.

NOTE – Fabric structure may limit SCN to a Fabric Region.

When a state change occurs, initiative for SCN shall be established in the Fabric only for N_Ports currently logged into the Fabric, which are capable of communicating with the N_Port which changed state, and which have registered to be notified.

Protocol:

State Change Notification request Sequence
No reply Sequence

Format: FT-1

Addressing: The S_ID field designates the N_Port detecting or reporting the Login state change. The D_ID field designates the destination N_Port of the Login state change notification.

Payload: The format of the Payload is shown in table 124.

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Affected N_Port ID page. The rightmost two bits shall be zeros.

Table 124 – SCN payload

Item	Size -Bytes
x‘22’ = SCN Command code	1
x‘04’ = Page Length	1
Payload Length	2
Affected N_Port ID pages	4-max

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the SCN Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 8, and less than or equal to 65 532.

Affected N_Port ID page(s): Words 1:max of the SCN Payload contain one or more Affected N_Port ID pages. Each Affected N_Port ID page contains a single N_Port ID identified as sustaining a state change. The format of SCN Affected N_Port ID pages is described in table 125.

Table 125 – SCN affected N_Port ID page format

Item	Word	Bit
Affected N_Port ID page	0	31-0
Reserved	0	31-24
Affected N_Port ID	0	23-0

The Affected N_Port ID shall not be all ones or all zeros nor shall the address match the D_ID of the SCN frame. If word 1, byte 0 of the Payload is not zero, the request is discarded, and a link-protocol error is detected. If an SCN request is received and the Affected N_Port ID is all ones, all zeros, or matches the D_ID of the SCN request, the request is discarded, and a link-protocol error is detected.

21.14 Test process login state

The Test Process Login State (TPLS) Extended Link Service request shall be used to determine whether a image pairs are established for the image pairs specified by the combination of the S_ID || Originator Process_Associator || D_ID || Responder Process_Associator. Upon receiving a TPLS request, the receiving N_Port checks whether it has an image pair established for each specified image.

TPLS verifies the Login state for the source N_Port at the destination N_Port.

NOTE – For example, an SB CH can use TPLS to determine CU Login state for that CH and an SB CU can use TPLS to determine CH Login state for that CU.

The Accept reply Sequence confirms the successful completion of the TPLS function and indicates whether or not an image pair is established for the source specified by the S_ID and image pair(s) contained in the Payload. The Payload of the Accept reply indicates the state of the image pair.

Protocol:

Test Process Login State request Sequence
Accept reply Sequence

Format: FT-1

Addressing: The S_ID field designates the source N_Port associated with the image pair. The D_ID field designates the destination N_Port associated with the image pair.

Payload: The format of the Payload is shown in table 126.

Table 126 – TPLS payload

Item	Size -Bytes
x‘23’ = TPLS Command code	1
x‘10’ = Page Length	1
Payload Length	2
Image Pair ID pages	16-max

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Image Pair ID page. The rightmost two bits shall be zeros.

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the TPLS Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

Image pair ID page(s): Words 1:max of the TPLS Payload contain one or more Image Pair

Table 127 – TPLS image pair ID page format

Item	Word	Bit
Image Pair ID page	0-3	31-0
Reserved	0	31-16
Originator Process_ Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Reserved	0	13-0
Originator Process_ Associator	1	31-0
Responder Process_ Associator	2	31-0
Reserved	3	31-0

ID pages. Each Image Pair ID page contains parameters required to identify a single image pair.

The format of Image Pair ID pages is described in table 127.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

0 = not meaningful

1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

0 = not meaningful

1 = meaningful

Reply link service sequence:

Link Service Reject (LS_RJT)
signifies rejection of the TPLS request
Accept (ACC)
signifies successful completion of the TPLS request
- Accept Payload
The format of the Accept Payload is shown in table 128.

Page length: Byte 1 of word 0 contains an 8-bit unsigned binary integer that specifies the length of each Image Pair ID page. The rightmost two bits shall be zeros.

Table 128 – TPLS accept payload

Item	Size -Bytes
x'02' = ACC Command code	1
x'10' = Page Length	1
Payload Length	2
TPLS Response pages	16-max

Payload length: Bytes 2-3 of word 0 contain a 16-bit unsigned binary integer that specifies the length of the TPLS Accept Payload. The rightmost two bits shall be zeros. The value specified shall be greater than or equal to 20, and less than or equal to 65 532.

TPLS response page(s): Words 1:max of the TPLS Payload contain one or more Image Pair ID pages. Each TPLS Response page contains TPLS response information associated with a single image pair.

The number of Service Parameter pages and the Payload Length field of the TPLS Accept reply Sequence shall match the number of

Table 129 – TPLS response page format

Item	Word	Bit
TPLS Response page	0-3	31-0
Reserved	0	31-16
Originator Process_ Associator Validity	0	15
Responder Process_ Associator Validity	0	14
Reserved	0	13-12
Response Code (see table 130)	0	11-8
Reserved	0	7-0
Originator Process_ Associator	1	31-0
Responder Process_ Associator	2	31-0
Reserved	3	31-1
Image Pair State	3	0

Service Parameter pages and the Payload Length field of the TPLS request Sequence.

The format of TPLS Response pages is described in table 129.

- **Word 0, Bit 15 - Originator Process_ Associator validity**

  0 = not meaningful

  1 = meaningful

- **Word 0, Bit 14 - Responder Process_ Associator validity**

  0 = not meaningful

  1 = meaningful

- **Word 0, Bits 11-8 - Response code**

  The Response code field contains an encoded binary value indicating the result of the PRLI request and the status of the image pair. The meanings of the encoded Response code values are shown in table 130.

- **Word 3, Bit 0 - Image pair state**

  1 = image pair established

  0 = image pair not established

Table 130 – TPLS accept response code

Encoded Value Wd 0, Bits 11-8	Description
0000	Reserved
0001	Request executed
0010:0110	Reserved
0111	The destination N_Port is unable to process multiple page PRLI requests. The PRLI request may be retried as a single page request.
1000:1111	Reserved

21.15 Alias Extended link services

The Alias Server requires the definition of other Link Services commands that are not currently defined in FC-PH or FC-FG. The following sections (21.16 and 21.17) define these requests and their responses.

21.16 Fabric Link Services for Alias

The following additional Fabric Controller Link Services are defined:

- Get Alias_ID (GAID)
- Fabric Activate Alias_ID (FACT)
- Fabric Deactivate Alias_ID (FDACT)

21.16.1 Get Alias_ID (GAID)

This request is sent to the Fabric Controller by Alias Server (see figure 90) to request a unique alias address identifier to be associated with the Alias Group indicated in the passed Alias_Token. The payload of the request contains, among other parameters, the Alias_Token of the Alias Group to be created. The format of the payload is shown in table 131.

Table 131 – Get Alias_ID Payload

Item	Size (Bytes)
hex ‘30000000’	4
Alias_Token	12
Alias_SP	80
NP_List_Length	4
NP_List(1)	4
NP_List (2 to n-1)	(n-2) x 4
NP_List(n)	4

Alias Group Token (Alias_Token): This is the token identifying the Alias Group. See FC-GS for a description of the Alias_Token.

Alias Group Service Parameters (Alias_SP):

The Alias_SP defines the Service Parameters to be used for all operations with this Alias Group. The Service Parameters are passed in the format defined in FC-PH, although only the Common Service Parameters and the appropriate Class Service Parameters are actually used.

NOTE – This field is used by the Fabric Controller to ensure that the Service Parameters to be used by the Alias Group are not in conflict with the Fabric Service Parameters.

N_Port List Length (NP_List_Length): The NP_List_Length specifies the number of entries in the following NP_List.

N_Port List (NP_List): The NP_List contains one entry for each N_Port address identifier to be included in the Alias Group. The N_Port address identifier shall be right-aligned within the NP_List entry. That is, the high-order byte of the entry shall be ignored and the low-order 3 bytes shall contain the N_Port address identifier. The N_Port address identifier shall not be an Alias_ID.

NOTE – The Fabric Controller uses the NP_List to determine whether it can support an Alias Group composed of these N_Ports. For example, if the Fabric cannot support an Alias Group that spans Domains, and the NP_List contains N_Ports in multiple Domains, the request may be rejected. Alternatively, the Fabric may form an Alias Group from the N_Ports within a single Domain.

The format of the payload for the LS_ACC indicating that this request has been successfully completed is shown in table 132.

Table 132 – Get Alias_ID Accept Payload

Item	Size (Bytes)
hex ‘02000000’	4
Alias_ID	4

Alias Group Identifier (Alias_ID): This is the alias address identifier that the N_Port will now recognize, in addition to its native address identifier.

An LS_RJT shall be returned if an Alias_ID cannot be generated. A Reason Code of “Unable to Perform Command Request” shall be generated, along with one of the Reason Explanations in table 91.

21.16.2 Fabric Activate Alias_ID (FACT)

This request is sent to a Fabric by Alias Server (see figure 90) to cause it to assign the passed Alias_ID as an alias for the passed N_Ports. The payload of the request contains, among other parameters a list of the N_Ports for which the alias identifier is to be recognized and the Alias_ID to be recognized as an alias identifier. The format of the payload is shown in table 133.

Table 133 – Fabric Activate Alias_ID Payload

Item	Size (Bytes)
hex ‘31000000’	4
Alias_ID	4
NP_List_Length	4
NP_List(1)	4
NP_List (2 to n-1)	(n-2)*4
NP_List(n)	4

Alias Group Identifier (Alias_ID): This is the alias address identifier that the Fabric shall recognize for the N_Ports in the list, in addition to their native address identifiers.

N_Port List Length (NP_List_Length): The NP_List_Length specifies the number of entries in the following NP_List.

N_Port List (NP_List): The NP_List contains one entry for each N_Port for which the Alias_ID is to be assigned as an alias identifier. The N_Port address identifier shall be right-aligned within the NP_List entry. That is, the high-order byte of the entry shall be ignored and the low-order 3 bytes shall contain the N_Port address identifier.

The format of the payload for the LS_ACC indicating that this request has been successfully completed is shown in table 134.

Table 134 – Activate Alias_ID Accept Payload

Item	Size (Bytes)
hex ‘02000000’	4

An LS_RJT shall be returned if an Alias_ID cannot be activated as an alias identifier. A Reason Code of “Unable to Perform Command Request” shall be generated, along with one of the Reason Explanations in table 91.

21.16.3 Fabric Deactivate Alias_ID (FDACT)

This request is sent to the Fabric Controller by the Alias Server (see figure 90) to request that it deassign the indicated Alias_ID as an Alias Group alias identifier for the passed N_Ports. The payload of the request contains, among other parameters, the Alias_ID to be deactivated, and the list of N_Ports for which this Alias_ID is to be deactivated. When there are no longer any N_Ports for which the Alias ID is active, the Fabric Controller shall free up the Alias ID for subsequent reassignment. The format of the payload is shown in table 135.

Table 135 – Fabric Deactivate Alias_ID Payload

Item	Size (Bytes)
hex ‘32000000’	4
Alias_ID	4
NP_List_Length	4
NP_List(1)	4
NP_List (2 to n-1)	(n-2)*4
NP_List(n)	4

Alias Group ID (Alias_ID): This is the alias identifier to be deactivated.

N_Port List Length (NP_List_Length): The NP_List_Length specifies the number of entries in the following NP_List.

N_Port List (NP_List): The NP_List contains one entry for each N_Port for which the MG_ID is to be deactivated as an alias identifier. The N_Port address identifier shall be right-aligned within the NP_List entry. That is, the high-order byte of the entry shall be ignored and the low-order 3 bytes shall contain the N_Port address identifier.

The format of the payload for the LS_ACC indicating that this request has been successfully completed is shown in table 136.

Table 136 – Fabric Deactivate Alias_ID Accept Payload

Item	Size (Bytes)
hex ‘02000000’	4

An LS_RJT shall be returned if the Alias_ID cannot be deactivated as an alias identifier. A Reason Code of “Unable to Perform Command Request” shall be generated, along with one of the Reason Explanations in table 91.

21.17 N_Port Link Services for Alias

The following additional N_Port Link Services are defined:

- N_Port Activate Alias_ID (NACT)
- N_Port Deactivate Alias_ID (NDACT)

21.17.1 N_Port Activate Alias_ID (NACT)

This request is sent to an N_Port by Alias Server (see figure 90) to cause it to assign the passed alias address identifier as an alias. The payload of the request contains, among other parameters, the Alias_Token of the Alias Group which the N_Port will be joining, the Service Parameters associated with that Alias Group, and the alias address identifier to be assigned to the N_Port. The format of the payload is shown in table 137.

Table 137 – N_Port Activate Alias_ID Payload

Item	Size (Bytes)
hex ‘33000000’	4
Alias_Token	12
Alias_ID	4
Alias_SP	80

Alias Group Token (Alias_Token): This is the token identifying the Alias Group. It contains the Alias_Class and Alias_Qualifier of the Alias Group the N_Port is being requested to join. The N_Port validates that it supports that particular Alias Class. See FC-GS for a description of the Alias_Token.

Alias Group Identifier (Alias_ID): This is the alias address identifier that the N_Port will now recognize, in addition to its native address identifier.

Alias Group Service Parameters (Alias_SP)

The Alias_SP defines the Service Parameters to be used for all operations with this Alias Group. The Service Parameters are passed in the format defined in FC-PH.

NOTE – These Service Parameters may differ from those passed during Login.

For Multicast Groups, only the Common Service Parameters and Class 3 Service Parameters are actually used. For Hunt Groups, all Service Parameters may be used.

These Service Parameters are used to perform an implicit Login among the members of the Alias Group.

If the N_Port can successfully perform the following functions, it shall transmit an LS_ACC to notify the Alias Server that it:

- Can support the Alias_Class in the Alias_Token.
- Can support the Alias Group Service Parameters specified in Alias_SP.
- Has assigned the passed Alias_ID as an alias for this N_Port.

If the N_Port cannot perform all of the above functions, it shall send an LS_RJT as a reply.

The format of the payload for the LS_ACC indicating that this request has been successfully completed is shown in table 138.

Table 138 – N_Port Activate Alias_ID Accept Payload

Item	Size (Bytes)
hex ‘02000000’	4

An LS_RJT shall be returned if the Alias_ID cannot be activated as an alias identifier. A Reason Code of “Unable to Perform Command Request” shall be generated, along with one of the Reason Explanations in table 91.

21.17.2 N_Port Deactivate Alias_ID (NDACT)

This request is sent to an N_Port by Alias Server (see figure 90) to cause it to deactivate the passed Alias_ID as an alias identifier. The payload of the request contains, among other parameters, the alias address identifier to be deactivated by the N_Port. The format of the payload is shown in table 139.

Table 139 – N_Port Deactivate Alias_ID Payload

Item	Size (Bytes)
hex ‘34000000’	4
Alias_ID	4

Alias Group Identifier (Alias_ID): This is the alias address identifier that the N_Port will no longer recognize.

If the N_Port can successfully deactivate the Alias_ID, it shall transmit an LS_ACC to notify the Alias Server.

The format of the payload for the LS_ACC indicating that this request has been successfully completed is shown in table 140.

Table 140 – N_Port Deactivate Alias_ID Accept Payload

Item	Size (Bytes)
hex ‘02000000’	4

An LS_RJT shall be returned if the Alias_ID cannot be deactivated as an alias identifier. A Reason Code of “Unable to Perform Command Request” shall be generated, along with one of the Reason Explanations in table 91.

21.18 Class 4 Extended Link Services

Class 4 Service requires the definition of Link Service commands that are not currently defined in FC-PH. The following defines these Extended Link Service commands for Class 4.

The following Extended Link Service requests shall be supported for Class 4:

Quality of Service Request (QoSR)

Read Virtual Circuit status (RVCS).

21.18.1 Quality of Service Request (QoSR)

The QoSR Extended Link Service Sequence requests setup of a Class 4 circuit and that the requested level of service be granted by the Quality of Service Facilitator (QoSF) for this Class 4 circuit. The N_Port requesting a Class 4 circuit is the circuit initiator (CTI), the target N_Port is the circuit recipient (CTR). The CTR shall grant the setup of the Class 4 circuit before the Class 4 circuit setup is granted to the CTI (see figure 86). A QoSR transaction from the QoSF to the CTR serves to obtain acceptance of the CTR (see 34.2.4.1).

Protocol:
Quality of Service Request (QoSR) request Sequence
Accept (ACC) reply Sequence

Format: FT_1

Addressing: When transmitted by a CTI, the S_ID field designates the source N_Port requesting the Class 4 circuit. The D_ID field designates the QoSF (hex 'FFFFF9').

When transmitted by the QoSF, the S_ID field designates the QoSF. The D_ID field designates the CTR Address Identifier.

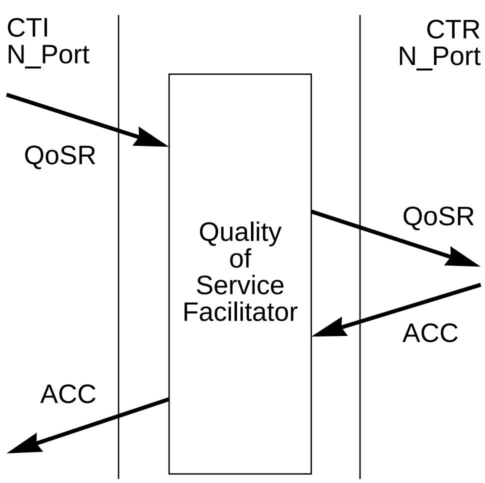


Figure 86 – QoSR Sequence flow

Payload: The format of the Payload is shown in table 141.

CTI VC_ID: The CIVC_ID for the CTI. The CTI shall choose a currently unassigned VC_ID value in the range of hex '01' through hex 'FE'. This same value shall be in the request from the QoSF to the CTR.

CTI Address Identifier: The Address Identifier in the originating request S_ID field. This same value shall be in the request from the QoSF to the CTR.

The CTI address identifier may be different from the S_ID in the QoSR from the CTI. However, the CTI address identifier shall be a valid, native or alias address identifier for the CTI.

Table 141– QoSR Payload

Item	Size Bytes
hex '40000000'	4
Reserved	4
CTI VC_ID	1
CTI Address Identifier	3
CTI Maximum Bandwidth (B/s)	4
CTI Minimum Bandwidth (B/s)	4
CTI Maximum Delay (µs)	4
CTI VC Data Field Size	4
Reserved	12
CTR VC_ID	1
CTR Address Identifier	3
CTR Maximum Bandwidth (B/s)	4
CTR Minimum Bandwidth (B/s)	4
CTR Maximum Delay (µs)	4
CTR VC Data Field Size	4
Reserved	12
Reserved	1
Live VC credit limit	1
Class 4 end-to-end credit	2

The CTI address identifier shall be the address identifier used as the S_ID or D_ID on all Class 4 frames carried on the Class 4 circuit setup by this QoSR.

CTI Maximum Bandwidth (B/s): The maximum bandwidth in bytes per second (B/s), from the CTI to the CTR, the CTI requires for this VC. This same value shall be in the request from the QoSF to the CTR.

The Fabric may enforce the maximum bandwidth limit. An N_Port may use end-to-end flow control to further limit the available bandwidth.

CTI Minimum Bandwidth (B/s): The minimum bandwidth in bytes per second (B/s) acceptable from the CTI to the CTR. This same value shall be in the request from the QoSF to the CTR.

The Fabric shall guarantee this bandwidth if in any 1 ms interval, the average frame size satisfies the following condition:

$$\overline{FrameSize} \geq \frac{ResponderVCDataFieldSize + 180}{3}$$

CTI Maximum Delay (µs): The maximum acceptable delay in microseconds from the CTI to the CTR. This same value shall be in the request from the QoSF to the CTR. The CTI maximum delay value shall satisfy the following relationship:

$$E\_D\_TOV > 2\,(CIMD + CRMD)$$

where

CIMD CTI maximum delay

CRMD CTR maximum delay

CTI VC Data Field Size: The maximum acceptable frame size in bytes from the CTI to the CTR. This value shall not exceed the minimum value resulting from Fabric Login and N_Port Login with the CTR. This value shall not be less than 256 and shall be a multiple 4. This same value shall be in the request from the QoSF to the CTR.

CTR VC_ID: When sent from the CTI, this field shall be zero. This same value shall be in the request from the QoSF to the CTR.

CTR Address Identifier: The Address Identifier of the CTR for this Class 4 circuit. This same value shall be in the request from the QoSF to the CTR.

CTR Maximum Bandwidth (B/s): The maximum bandwidth in bytes per second (B/s), from the CTR to the CTI. This same value shall be in the request from the QoSF to the CTR.

The Fabric may enforce the maximum bandwidth limit. An N_Port may use end-to-end flow control to further limit the available bandwidth.

CTR Minimum Bandwidth (B/s): The minimum bandwidth in bytes per second (B/s) acceptable from the CTR to the CTI. This same value shall be in the request from the QoSF to the CTR.

The Fabric shall guarantee this bandwidth if in any 1 ms interval, the average frame size satisfies the following condition:

$$\overline{FrameSize} \geq \frac{OriginatorVCDataFieldSize + 180}{3}$$

CTR Maximum Delay (µs): The maximum acceptable delay in microseconds from the CTR to the CTI. This same value shall be in the request from the QoSF to the CTR. The CTR maximum delay value shall satisfy the following relationship:

$$E\_D\_TOV > 2\,(CIMD + CRMD)$$

where

CIMD CTI maximum delay

CRMD CTR maximum delay

CTR VC Data Field Size: The maximum acceptable frame size in bytes from the CTR to the CTI. This value shall not exceed the minimum value resulting from Fabric Login and N_ Port Login with the CTR. This value shall not be less than 256 and shall be a multiple 4. This same value shall be in the request from the QoSF to the CTR.

Live VC credit limit: The VC credit limit for the VC in the live state shall be set by the Fabric in the QoSR from the QoSF to the CTR.

Class 4 end-to-end Credit: For the CTR, this shall represent the maximum Class 4 end-to-end Credit limit as well as the Class 4 end-to-end Credit for a dormant Class 4 circuit. This same value shall be in the request from the QoSF to the CTR.

Reply Link Service Sequence (Reply to QoSR):

Service Reject (LS_RJT)

signifies rejection of the QoSR command

Accept (ACC)

From the CTR to the QoSF - signifies that the CTR has accepted the QoS parameters.

From the QoSF - signifies that the Fabric has accepted the QoS parameters and has setup the Class 4 circuit. The QoSF shall await the ACC from the CTR before it transmits the ACC to the CTI.

Accept Payload

The format of the Accept Payload is shown in table 141.

CTI VC_ID: The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTI Address Identifier: The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

Table 142– QoSR Accept Payload

Item	Size Bytes
hex ’02000000’	4
Reserved	4
CTI VC_ID	1
CTI Address Identifier	3
CTI Maximum Bandwidth (B/s)	4
CTI Minimum Bandwidth (B/s)	4
CTI Maximum Delay (µs)	4
CTI VC Data Field Size	4
Reserved	12
CTR VC_ID	1
CTR Address Identifier	3
CTR Maximum Bandwidth (B/s)	4
CTR Minimum Bandwidth (B/s)	4
CTR Maximum Delay (µs)	4
CTR VC Data Field Size	4
Reserved	12
Reserved	1
Live VC credit limit	1
Class 4 end-to-end credit	2

CTI Maximum Bandwidth (B/s): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTI Minimum Bandwidth (B/s): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTI Maximum Delay (µs): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTI VC Data Field Size: The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTR VC_ID: The field is only meaningful in the ACC and shall contain the CRVC_ID for the CTR. The CTR N_Port shall choose an unused value in the range of hex ‘01' through hex ‘FE'. This same value shall be in the ACC from the QoSF to the CTI.

CTR Address Identifier: The value from the ACC from the CTR shall be in the ACC from the QoSF to the CTI. The value in the ACC from the CTR may be different from the value in the QoSR from the QoSF.

The CTR address identifier may also be different from the S_ID in the ACC from the CTR. However, the CTR address identifier shall be a valid, native or alias address identifier for the CTR.

The CTR address identifier shall be the address identifier used as the S_ID or D_ID on all Class 4 frames carried on the Class 4 circuit setup by this ACC.

CTR Maximum Bandwidth (B/s): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTR Minimum Bandwidth (B/s): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTR Maximum Delay (µs): The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

CTR VC Data Field Size: The value from the request shall be in the ACC from the CTR to the QoSF and from the QoSF to the CTI.

Live VC credit limit: The VC credit limit for the VC in the live state shall be set by the Fabric in the ACC from the QoSF to the CTI.

Class 4 end-to-end Credit: For the CTI, this shall represent the maximum Class 4 end-to-end Credit limit as well as the Class 4 end-to-end Credit for a dormant Class 4 circuit. The value in the ACC from the CTR shall be in the ACC from the QoSF to the CTI.

21.18.2 Read VC status (RVCS)

Read VC status (RVCS).

Protocol:

Read VC status Sequence

Accept (ACC) reply Sequence

Format: FT_1

Addressing: The S_ID shall designate the source N_Port requesting the VC status. The D_ID field shall designate the QoSF (hex ‘FFFFF9').

Payload: The format of the payload is shown in table 143.

Reply Link Service Sequence (Reply to RVCS):

Service Reject (LS_RJT)

Signifies rejection of the RVCS command

Accept (ACC)

Signifies that the QoSF has completed the request

Accept payload

The format of the Accept payload is shown in table 144.

The accept Sequence contains one entry for each Class 4 circuit setup with the designated N_Port. The format of the Class 4 VC status block entry is shown in table 145.

21.18.3 Link Service Reject (LS_RJT)

FC-PH-2 Table 91 shows additional LS_RJT reason code explanations applicable to Class4. The LS_RJT reason code explanations in FC-PH table 91 also apply to Class 4:

Table 143– RVCS Payload

Item	Size Bytes
hex ‘41000000’	4
Reserved	1
N_Port Identifier	3

Table 144– RVCS Accept Payload

Item	Size Bytes
hex ‘02000000’	4
Reserved	4
Class 4 VC status block (see table 145), where N = number of Class 4 circuits for the designated N_Port.	N*8

Table 145– Class 4 VC status block entry

Item	Size Bytes
CTI VC_ID	1
CTI Address Identifier	3
CTR VC_ID	1
CTR Address Identifier	3

21.19 Enhanced Login/Logout Link Service

21.19.1 Discover N_Port/F_Port Parameters (PDISC/FDISC)

The PDISC/FDISC Extended Link Service requests shall transfer Service Parameters from the initiating N_Port to the N_Port or F_Port associated with the Destination Identifier without affecting the operating environment between the two ports. In the case of PDISC, this means the exchange of Service Parameters without terminating open Sequences or open Exchanges. In the case of FDISC, this means the exchange of Service Parameters without establishing service with the fabric or affecting the internal configuration of the fabric.

The payload in PDISC and the ACC are, except for the LS_Command, identical to the payloads in PLOGI and the ACC. The payload in FDISC and the ACC are, except for the LS_Command, identical to the payloads in FLOGI and the ACC. PDISC and FDISC LS_Command codes are x'50' and x'51' (see Table 61). The interchange of FDISC or PDISC information shall not modify the operating environment or Service Parameters between the two ports. PDISC and FDISC may be transmitted in any class of service. Class 1, 2, 3, or 4 PDISC and FDISC transmission may be attempted in any order.

A response shall not be sent to a PDISC if an LR or FC-AL initialization occurs before the ACC is sent. An ACC for a PDISC shall be ignored if an LR or FC-AL initialization occurred between the PDISC and the ACC.

Table 146 summarizes the responses to FDSIC/PDISC when the receiving Port is logged in or not logged in.

Table 146– Response summary to FDISC/PDISC

ELS command	Port Status	N_Port	F_Port	
		Class 1, 2, 3, 4	**Class 1, 2, or 4**	**Class 3**
FDISC	Logged in	LS_RJT [1]	ACC	ACC
	Not Logged in	LS_RJT [1]	F_RJT [2]	Discard
PDISC	Logged in	ACC	F_RJT [2]	Discard
	Not Logged in	LS_RJT [1]	F_RJT [2]	Discard

[1] LS_RJT may be issued with a reason code = "unable to supply requested data"
[2] P_RJT may be issued with a reason code = "Login required"

If an unintelligent drive receives a PDISC from an N_Port with which it has not logged in or receives an FDISC, it may choose to issue a Logout or OLS.

21.19.2 Discover Address (ADISC)

The ADISC Extended Link Service request shall exchange addresses and identifiers of communicating N_Ports.

A Hard Address is the 24 bit NL_Port Identifier which consists of
- the 8 bit Domain address in the most significant byte,
- the 8 bit Area address in the next most significant byte, and
- the 8 bit AL_PA which an NL_Port attempts to acquire in the LIHA sequence during FC-AL initialization in the least significant byte (see FC-AL).

If an NL_Port does not have a hard address, or if a port does not have FC-AL capability, it shall report zeroes in this field.

Port Name in ADISC is the 8-byte Port Name of the Originator and Port Name in ACC to ADISC is the 8-byte Port Name of the Responder.

Node Name in ADISC is the 8-byte Node Name of the Originator and Node Name in ACC to ADISC is the 8-byte Node Name of the Responder.

N_Port ID is the 24-bit S_ID used in the header of the ADISC frame (for the Originator) or ACC frame (for the Responder).

A response shall not be sent to a ADISC if an LR or FC-AL initialization occurs before the ACC is sent. An ACC for a ADISC shall be ignored if an LR or FC-AL initialization occurred between the ADISC and the ACC.

Table 147 – ADISC Payload

Item	Size -Bytes
x‘52000000’ = ADISC Command code	4
Reserved	1
Hard Address of Originator	3
Port Name of Originator	8
Node Name of Originator	8
Reserved	1
N_Port ID of Originator	3

Table 148 – ADISC ACC Payload

Item	Size -Bytes
x‘02000000’ = ACC Command code	4
Reserved	1
Hard Address of Responder	3
Port Name of Responder	8
Node Name of Responder	8
Reserved	1
N_Port ID of Responder	3

21.19.3 Third Party Process Logout (TPRLO)

The Third Party Process Logout (TPRLO) Extended Link Service request shall be used to invalidate the operating environments between the specified image(s) at the recipient N_Port and the specified image(s) in the specified N_Port(s) which have performed Process Login with the recipient N_Port for the specified TYPE.

TPRLO has the same effect on the recipient N_Port as if all specified Third Party Originators performed PRLO with the recipient N_Port using the specified Process Associators.

Third Party Originator Process Associator validity, if set to 1, indicates that word 2, bits 31-0 are meaningful.

The Global bit, if set to 1, specifies that all image pairs for all N_Ports with which Process Login has been performed be removed from the recipient N_Port for the specified TYPE. If the Global bit is set to 1, only one logout parameter page shall be transmitted, and only the TYPE code and TYPE code extension fields shall have meaning (aside from the Global bit itself). If the Global bit is set to 1, it frees all resources associated with the establishment of all image pairs of the specified TYPE at the recipient N_Port.

Third Party Originator N_Port ID validity, if set to 1, indicates that word 1, bits 23-0 are meaningful.

The Third Party Originator Process_Associator specifies the Process Associator associated

Table 149 – TPRLO Payload

Item	Size -Bytes
x'24 ' = TPRLO Command code	1
x'10' = Page Length	1
Payload length	2
Logout Parameter pages	20-max

Table 150 – TPRLO logout parameter page format

Item	word	Bit
TYPE Code or Common Service Parameters	0	31-24
TYPE Code Extension	0	23-16
Third Party Originator Process _Associator Validity	0	15
Responder Process_Associator Validity	0	14
Third Party Originator N_Port ID Validity	0	13
Global Process Logout	0	12
Reserved	0	11-0
Third Party Originator Process_Associator	1	31-0
Responder Process_Associator	2	31-0
Reserved	3	31-24
Third Party Originator N_Port ID	3	23-0
Reserved	4	31-0

Table 151 – TPRLO ACC Payload

Item	Size -Bytes
x'02 ' = ACC Command code	1
x'10' = Page Length	1
Payload length	2
Logout Parameter pages	20-max

with the image to be removed.

The Third Party Originator N_Port ID specifies the N_Port associated with the image to be removed.

All other fields have the same meaning as described in PRLO.

The minimum TPRLO Payload length is 24 bytes.

22 Classes of Service

Enhancements to Classes of Service from FC-PH are specified.

22.5 Class 4

Class 4 is a new Class of Service specified in FC-PH-2. Refer to Clause 34 for details on Class 4.

23 Login and Service Parameters

Enhancements to FC-PH Login and Service Parameters are specified. There are no other changes to FC-PH clause 23.

23.3 Fabric Login

Class 4 is not applicable to Fabric Login and shall not be used for Fabric Login. There are no changes to the FLOGI process specified in FC-PH 23.3.

The Class 4 N_Port Login service parameters used during Fabric Login (in FLOGI or ACC to FLOGI) shall be as shown in table 98.

The Fabric shall remove all Class 4 connections to an N_Port when a Fabric Login is performed by the N_Port.

23.3.3 SOF delimiters (FC-PH section revised)

Selection of the SOF delimiter for the FLOGI Sequence is based on the Classes of Service supported by the originating N_Port. Since the F_Port may not support all Classes of Service, the FLOGI Sequence may have to be retried with SOF delimiters for different Classes of Service. The FLOGI Sequence is transmitted and the appropriate action is specified in FC-PH table 94 or 95. If a Fabric Reject (F_RJT) with reason code "Class of Service not supported by entity at FFFFFE" has been received, another supported delimiter is attempted until the Login procedure is complete or until all supported delimiter types have been attempted. If an Accept Reply Sequence (ACC) is received, the validity bits in the Accept Sequence indicate which Classes of Service are supported by the Fabric.

23.4 N_Port Login

The addition of Class 4 and enhancements to other Classes of Service introduce additional choices to FC-PH-2 in performing N_Port Login.

The Class 4 default N_Port Login service parameters prior to N_port Login shall be as shown in table 98.

The state of Class 4 connections shall not be affected by end-to-end N_Port Login.

23.4.2 Explicit N_Port Login

The optional support of Class 1 and Buffered Class 1 by N_Ports and F_Ports results in various communication capabilities between them:

– A given N_Port pair may have neither buffered nor unbuffered Class 1 communication capability.

– A given N_Port pair may have the capability to communicate using only unbuffered Class 1.

– A given N_Port pair may have the capability to communicate using only buffered Class 1.

– A given N_Port may have the capability to communicate using Buffered or unbuffered Class 1 with different N_Ports.

The optional support for Class 4 by N_Ports and F_Port means that a pair of N_Ports may be able to communicate using Class 4.

For these reasons, the order of Classes tried for N_Port Login specified in FC-PH 23.4.3 is modified as follows:

a) If permitted by FLOGI, PLOGI shall be tried first using unbuffered Class 1 Service.

b) If unsuccessful, PLOGI shall be attempted, FLOGI permitting, in Buffered Class 1 Service.

c) If that is also unsuccessful, PLOGI shall be attempted, FLOGI permitting, in Class 2 Service.

d) If unsuccessful, PLOGI shall be attempted, FLOGI permitting, in Class 3 Service.

e) If unsuccessful, and permitted by FLOGI and if a Class 4 circuit can be established with the destination N_Port, PLOGI shall be attempted using Class 4 Service. If this attempt is also unsuccessful, then no communication is possible between the two N_Ports.

23.6 N_Port Service Parameters

Figure 59 provides the payload format of the FLOGI or PLOGI Extended Link Service commands and their respective ACC replies.

16 bytes	8 bytes	8 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes
Common Service Parameters	Port Name	Node or Fabric Name	Class 1 Service Parameters	Class 2 Service Parameters	Class 3 Service Parameters	Class 4 Service Parameters	Vendor Version Level

Figure 59 – FLOGI, PLOGI, or ACC Payload

23.6.1 N_Port Common Service Parameters

Table 98 provides a summary of N_Port Common Service Parameters based on FC-PH table 98.

Table 98 – N_Port Common Service Parameter Applicability

Service Parameter	Wd	Bits	N_Port Login Class (see 23.6.3)				Fabric Login Class (see 23.6.2)			
			1	2	3	4[5]	1	2	3	4[5]
FC-PH Version										
Highest Version	0	31-24	y[1]	y	y	y	y	y	y	y
Lowest Version	0	23-16	y	y	y	y	y	y	y	y
Buffer-to-Buffer Credit	0	15-0	y	y	y	n[2]	y	y	y	n
Common Features										
Continuously Increasing (C)	1	31	y	y	y	y	n	n	n	n
Random Relative Offset (O)	1	30	y	y	y	y	n	n	n	n
Vendor Version Level	1	29	y	y	y	y	y	y	y	y
N_Port/F_Port	1	28	y	y	y	y	y	y	y	y
Alternate BB_Credit Management (see 26.5)	1	27[a]	y	y	y	y	y	y	y	y
Buffer-to-Buffer Receive Data field Size	1	11-0	y	y	y	y	y	y	y	y
N_Port Total Concurrent Sequences	2	31-16	y	y	y	y	n	n	n	n
Relative Offset by Info Category	2	15-0	y	y	y	y	n	n	n	n
E_D_TOV [3, 4]	3	31-0	PTP	PTP	PTP	PTP	n	n	n	n

Notes

[1] “y” indicates yes, applicable.

[2] “n” indicates, not applicable.

[3] PTP indicates the applicability only to Point-to-Point.

[4] R_A_TOV and E_D_TOV provided by the F_Port as Fabric Common Service Parameters during Fabric Login are not shown in this table.

[5] Class 4 (see clause 34).

[a] See FC-PH-3 for additional bit to support the new use of E_D_TOV timer resolution.

23.6.2 N_Port Common Service parameters - Fabric Login

Enhanced N_Port Common Service parameters used during Fabric Login are illustrated in figure 60.

word	31 … 16	15 … 0
0	FC-PH / FC-PH-2 Version HHHHHHHH LLLLLLLL	Buffer-to-buffer Credit BBBBBBBB BBBBBBBB
1	Common Features r rV NA r r r r r r r r r rrr	Buffer-to-buffer Rcv Data Field size r r r r F F F F FFFF FFFF
2	Total Concurrent Sequences Reserved r r r r r r r r r r r r r r r r	Relative Offset by Info Category Reserved r r r r r r r r r r r r r r r r
3	Reserved r r r r r r r r t t t t t t t t	Reserved r r r r r r r r r r r r r r r r

Figure 60 – N_Port Common Service Parameters - Fabric Login

23.6.2.1 FC-PH-2 version

Table 99 specifies the hexadecimal values for low and high FC-PH-2 version levels indicated in word 0, bits 31 -16 of N_Port Common Service parameters.

Table 99 – FC-PH-2 Version

Hex value	Version level
00 - 09	See FC-PH Tables 99 or 100
0A - 0F	Reserved
10	FC-PH-2
11 - FF	Reserved

23.6.2.3 Common features

- **Word 1, bits 31 - 30**

Specified in FC-PH.

- **Word 1, bit 29 Vendor Specific (V)**

0 = Not valid

1 = Valid

Word 1 bit 29 indicates that the Vendor Version Level (fifth 16-byte group) contains valid information. If Word 1 bit 29 = 0, it indicates that the Vendor Version Level field is not meaningful.

- **Word 1, bit 28**

Specified in FC-PH.

- **Word 1, bit 27 BB_Credit Management (A)**

0 = BB_Credit management specified in FC-PH 26.5.1 through 26.5.9.
1 = Alternate BB_Credit management (see 26.5)

The BB_Credit Management field specifies the type of BB_Credit Management to be used.

- **Word 1, bits 26 -16**

Specified in FC-PH

23.6.3 N_Port Common Service parameters - N_Port Login

Enhanced N_Port Common Service parameters used during N_Port Login are illustrated in figure 61.

word	31 – 16	15 – 0
0	FC-PH / FC-PH-2 Version HHHHHHHH LLLLLLLL	Buffer-to-buffer Credit BBBBBBBB BBBBBBBB
1	Common Features COVNArrr r D r r r r r r	Buffer-to-buffer Rcv Data Field size r r r r FFFF FFFF FFFF
2	Total Concurrent Sequences r r r r r r r r TTTTTTTT	Relative Offset by Info Category DDDDDDDD DDDDDDDD
3	E_D_TOV (Pt-to-Pt) t t t t t t t t t t t t t t t t	t t t t t t t t t t t t t t t t

Figure 61 – N_Port Common Service Parameters - N_Port Login

23.6.3.1 FC-PH-2 version

Same as 23.6.2.1.

23.6.3.3 Common features

- **Word 1, bits 31 - 28**

Specified in FC-PH.

- **Word 1, bit 27 BB_Credit Management (A)**

0 = BB_Credit management specified in FC-PH 26.5.1 through 26.5.9.
1 = Alternate BB_Credit management (see 26.5)

The BB_Credit Management field specifies the type of BB_Credit Management to be used.

- **Word 1, bit 22 Dedicated Simplex (D)**

0 = Dedicated Simplex not supported by the N_Port

1 = Dedicated Simplex supported by the N_Port

- **Word 1, bits 26 - 23, 21 -16** [b]

Specified in FC-PH.

b. See FC-PH-3 for additional bit (word 1, bit 26) to support the new use of E_D_TOV timer resolution.

Table 101 – N_Port Class Service Parameter Applicability

Service Parameter	Wd	Bits	N_Port Login Class (see 23.6.8)				Fabric Login Class (see 23.6.7)			
			1	2	3	4[3]	1	2	3	4[3]
Class Validity										
Valid = 1	0	31	y[1]	y	y	y	y	y	y	y
Service Options										
Intermix Mode	0	30	y	n[2]	n	n	y	n	n	n
Stacked Connect-Request (see clause 36)	0	29-28	n	n	n	n	n	n	n	n
Sequential Delivery	0	27	n	n	n	n	n	y	y	n
Dedicated Simplex (see clause 33)	0	26	y	n	n	n	y	n	n	n
Camp-On (see clause 35)	0	25	n	n	n	n	y	n	n	n
Buffered Class 1 (see clause 37)	0	24	y	n	n	n	y	n	n	n
Initiator Control										
X_ID Reassignment	0	15-14	y	y	n	y	n	n	n	n
Initial Responder Processes_Associator	0	13-12	y	y	y	y	n	n	n	n
ACK_0 capable	0	11	y	y	n	y	n	n	n	n
ACK_N capable	0	10	y	y	n	y	n	n	n	n
ACK generation assistance (see 20.5)	0	9	y	y	n	y	n	n	n	n
Data compression capable (see clause 38)	0	8	y	y	y	y	n	n	n	n
Data compression History buffer size (see clause 38)	0	7-6	y	y	y	y	n	n	n	n
Recipient Control										
ACK_0 capable	1	31	y	y	n	y	n	n	n	n
ACK_N capable	1	30	y	y	n	y	n	n	n	n
X_ID Interlock	1	29	y	y	n	y	n	n	n	n
Error Policy Supported	1	28-27	y	y	y	y	n	n	n	n
Categories per Sequence	1	25-24	y	y	y	y	n	n	n	n
Data decompression capable (see clause 38)	1	23	y	y	y	y	n	n	n	n
Data compression History buffer size (see clause 38)	1	22-21	y	y	y	y	n	n	n	n
Reserved - Fabric Specific	1	19-16	y	y	y	y	y	y	y	y
Receive Data Field Size	1	15-0	y	y	y	y	n	n	n	n
Concurrent Sequences	2	31-16	y	y	y	y	n	n	n	n
N_Port End-to-End Credit	2	14-0	y	y	n	n	n	n	n	n
Open Sequences per Exchange	3	31-16	y	y	y	y	n	n	n	n

Notes -
[1] “y” indicates yes, applicable
[2] “n” indicates, not applicable
[3] Class 4 (see clause 34)

See FC-PH for all items without cross-references.

23.6.6 N_Port Class Service Parameters

Table 101 provides a summary of N_Port Class Service Parameters based on FC-PH table 101.

23.6.7 N_Port Class Service Parameters – Fabric Login

Only changes to FC-PH 23.6.7 are specified.

23.6.7.1 Class validity

- **Word 0, Bit 31 - Class validity**

  0 = Invalid (Class not supported)

  1 = Valid (Class supported)

The Class validity bit shall indicate whether this Class is supported or not. If the Class validity bit is zero, it indicates that this group or set of sixteen bytes shall be ignored. If the Class validity bit is one, it indicates that this Class shall be supported. The Class shall be identified based on Class 1 = first sixteen-byte group, Class 2 = second sixteen-byte group, Class 3 = third sixteen-byte group, and Class 4 = fourth sixteen-byte group.

NOTE – Service Parameter options specify FC-2 capability. The Link Service may further limit values supplied during Login as specified by individual Upper Level Protocols.

word	31 – 16	15 – 0
0	Service Options V I SS QDCB EEEEEEEE	Initiator Control DDDDDDDD DDDDDDDD
1	Recipient Control CCCCCCCC CCCCCCCC	Receive Data Field Size r r r r NNNN NNNNNNNN
2	Concurrent Sequences r r r r r r r r LLLLLLLL	N_Port End-to-end Credit 0MMM MMMM MMMMMMMM
3	Open Sequences per Exchange r r r r r r r r x x x x x x x x	Reserved r r r r r r r r r r r r r r r r

Figure 62 – N_Port Class Service Parameters - Fabric Login

23.6.7.2 Service options

Service options shall specify optional features of a Class of Service supported by the N_Port supplying the Service Parameters.

- **Word 0, bit 30**

Class 1, 2 and 3

No change from FC–PH.

Class 4

Word 0 bit 30 (Intermix Mode) has no meaning in Class 4 since the Fabric shall support Intermix on Class 4.

- **Word 0, bits 29 and 28**

Class 1

Word 0, bits 29 and 28 of the N_Port Class Service Parameters are reserved and not used during Fabric Login and set to zero:

  Word 0, bit29 = 0

  Word 0, bit 28 = 0

NOTE - Use of these bits has been changed from FC-PH.

Class 2 and 3

No change from FC–PH.

Class 4

Word 0, bits 29 and 28 (Stacked connect-request - Transparent and Lock-down modes) are reserved.

- **Word 0, bit 27**

Class 1, 2 and 3

No change from FC–PH.

Class 4

Word 0, bit 27 (Sequential Delivery)

Bit 27 has no meaning in Class 4 since the Fabric shall deliver frames in the order transmitted based on class of service.

- **Word 0, Bit 26 Dedicated Simplex (D)**

Class 1

Word 0, bit 26

  0=Dedicated Simplex not supported

  1=Dedicated Simplex supported

See clause 33.

Class 2, 3 and 4

Word 0, bit 26 is reserved.

- **Word 0, Bit 25 Camp-On (C)**

Class 1

Word 0, bit 25
0 = Camp-On not supported
1 = Camp-On supported

See clause 35.

Class 2, 3 and 4

Word 0, bit 25 is reserved.

- **Word 0, Bit 24 Buffered Class 1 (B)**

0 = Buffered Class 1 is not requested or Buffered Class 1 is not supported

1 = Buffered Class 1 is requested or Buffered Class 1 is supported

Buffered Class 1

Support for Buffered Class 1 Service is negotiated during Fabric Login. If an N_Port requests Buffered Class 1 Service and the attached F_Port supports it, only then is Buffered Class 1 Service enabled and available for use by the N_Port and Fabric. Table 101A summarizes the meaning of the combination of Word 0, bit 24.

Class 1, 2, 3 and 4

Word 0, bit 24 is reserved.

- **Word 0, Bit 23-16**

Reserved

Table 101A – Buffered Class 1 Login Service parameters

N_Port	F_Port	Description
0	0	Neither supports Buffered Class 1
0	1	Fabric is capable of supporting Buffered Class 1, Buffered Class 1 is nonfunctional
1	0	N_Port requests use of Buffered Class 1, Buffered Class 1 is nonfunctional
1	1	Fabric is capable of supporting Buffered Class 1 and N_Port request use of Buffered Class 1, Buffered Class 1 is functional (enabled)

23.6.7.3 Initiator Control

This field is not meaningful during Fabric Login.

23.6.7.4 Recipient Control

- **Word 1 bits 31 - 20**

These bits are not meaningful during Fabric Login.

- **Word 1 bits 19 - 16**

Reserved for Fabric Specific use.

23.6.7.5 Receive Data_Field Size

This field is not meaningful during Fabric Login.

23.6.7.6 Concurrent Sequences

This field is not meaningful during Fabric Login.

23.6.7.7 N_Port End-to-end Credit

This field is not meaningful during Fabric Login.

23.6.7.8 Open Sequences per Exchange

This field is not meaningful during Fabric Login.

23.6.8 N_Port Class Service Parameters - N_Port Login

Only changes to FC-PH 23.6.8 are specified. N_Port Class Service Parameters using during N_Port Login is illustrated in figure 62A.

word	bits 31 – 16	bits 15 – 0
0	Service Options V ISSQDCB r r r r r r r r	Initiator Control XXPPZNGC CCDDDDDD
1	Recipient Control ZNXLL r SS CCC r r r r r	Receive Data Field Size r r r rNNNNNNNNNNNN
2	Concurrent Sequences r r r r r r r r LLLLLLLL	N_Port End-to-end Credit 0 MMM MMMM MMMM MMMM
3	Open Sequences per Exchange r r r r r r r r x x x x x x x x	Reserved r r r r r r r r r r r r r r r r

Figure 62A – N_Port Class Service Parameters - N_Port Login

23.6.8.1 Class validity

- **Word 0, Bit 31 - Class validity (V)**

  0 = Invalid (Class not supported)

  1 = Valid (Class supported)

The Class validity bit shall indicate whether this Class is supported or not. If the Class validity bit is zero, it indicates that this group or set of sixteen bytes shall be ignored. If the Class validity bit is one, it indicates that this Class shall be supported. The Class shall be identified based on Class 1 = first sixteen-byte group, Class 2 = second sixteen-byte group, Class 3 = third sixteen-byte group, and Class 4 = fourth sixteen-byte group.

NOTE – Service Parameter options specify FC-2 capability. The Link Service may further limit values supplied during Login as specified by individual Upper Level Protocols.

23.6.8.2 Service options

Service options shall specify optional features of a Class of Service supported by the N_Port supplying the Service Parameters.

- **Word 0, bits 30 - 27 (ISSQ)**

Class 1, 2 and 3

No change from FC–PH.

Class 4

Word 0 bit 30 (Intermix Mode) has no meaning in Class 4 since the Fabric shall support Intermix on Class 4.

Word 0, bits 29 and 28 (Stacked connect-request - Transparent and Lock-down modes) are reserved.

Word 0, bit 27 (Sequential Delivery)-

Bit 27 has no meaning in Class 4 since the Fabric shall deliver frames in the order transmitted based on class of service.

- **Word 0, Bit 26 Dedicated Simplex (D)**

See clause 33.

Class 2, 3 and 4

Word 0, bit 26 is reserved.

- **Word 0, Bit 25 Camp-On (C)**

Class 1

Not meaningful in N_Port Login.

Class 2, 3 and 4

Word 0, bit 25 is reserved.

- **Word 0, Bit 24 - Buffered Class 1 (B)**

Word 0, Bit 24 is not meaningful for N_Port Login. See clause 37.

Class 1, 2, 3 and 4

Word 0, bit 24 is reserved.

- **Word 0, Bit 23-16**

Reserved

23.6.8.3 Initiator Control

- **Word 0, bits 15-10**

Class 1, 2 and 3

No change from FC–PH.

Class 4

- **Word 0, bit 15 -14 X_ID reassignment(XX)**

Same function as Class 1 and 2.

- **Word 0, bit 13-12 Initial Process_Associator (PP)**

Same function as Class 1 and 2.

- **Word 0 bit 11 ACK_0 capability (Z)**

Same function as Class 1 and 2.

- **Word 0, bit 10 ACK_N capability (N)**

Same function as Class 1 and 2.

Word, Bits	Description
0, 15 - 10	Specified in FC-PH 23.6.8.3
0, 9	ACK generation Assistance(G)
0, 8	Data compression capable (C)
0, 7-6	Data compression History buffer size (CC)
0, 5-0	Reserved (r)

Figure 65 – Initiator Ctl Flags (D)

- **Word 0, bit 9 - ACK generation assistance (G)**

Class 1, 2 and 3 (see 20.5)

Word 0, bit 9

0 = No ACK generation assistance is provided to Sequence Recipient.

1 = ACK generation assistance is provided to Sequence Recipient.

Class 4

Same function as Class 1 and 2.

- **Word 0, bit 8 - Data compression capable(C)**

Class 1, 2 and 3 (see 38.2)

Word 0, bit 8

= 0 Initiator does not have data com pression capability

=1 Initiator has data compression capability

Class 4

Same function as Class 1 and 2.

- **Word 0, bit 7 -6 - Data compression History buffer size (CC)**

Word 0, bits 7-6 are defined as shown in Table 101B.

Table 101B – History Buffer size (Initiator Control)	
Word 0 Bits 7-6	History buffer size
0 0	History buffer not available
0 1	512 bytes buffer is supported
1 0	1024 bytes buffer is supported
1 1	2048 bytes buffer is supported

Class 4

Same function as Class 1 and 2.

23.6.8.4 Recipient Control

- **Word 1, bits 31-24**

No change from FC–PH.

Word, Bits	Recipient Ctl Flags
1, 31 - 24	Specified in FC-PH 23.6.8.4 bit 31 - ACK_0 support (Z) bit 30 - ACK_N support (N) bit 29 - X_ID interlock (X) bits 28, 27 - Error policies (LL) bit 26 - Reserved (r) bits 25,24 - Categories (SS)
1, 23	Data decompression capability (C)
1, 22 - 21	Data decompression History buffer size (CC)
1, 20 - 16	Reserved (r)

Figure 66 – Recipient Ctl flags (C)

- **Word 1, bit 23- Data compression capable (C)**

Class 1, 2 and 3 (see 38.2)

Word 1, bit 23

0 = Recipient does not have data decompression capability
1 = Recipient has data decompression capability

Class 4

Same function as Class 1 and 2.

- **Word 1, bit 22 -21- Data compression History buffer size (CC)**

Class 1, 2 and 3 (see 38.2)

Word 1, bits 22 - 21 are defined as shown in Table 101C.

Table 101C – History Buffer size (Recipient Control)	
Word 1 Bits 22 - 21	History buffer size
0 0	History buffer not available
0 1	512 bytes buffer is supported
1 0	1024 bytes buffer is supported
1 1	2048 bytes buffer is supported

Class 4

Same function as Class 1 and 2.

- **Word 1, Bit 20-16**

Reserved

23.6.8.5 Receive Data_Field Size (N)

- **Word 1, Bits 15-0**

The Receive Data_Field Size is a binary value (bits 15-0) which specifies the largest Data_ Field Size for an FT_1 frame that can be received by the N_Port supplying the Service Parameters as a Sequence Recipient. Values less than 128 or greater than 2 112 are invalid. Values shall be a multiple of four bytes. An N_Port shall support a Data Field size of at least 128 bytes, however, a minimum of 256 bytes is recommended.

23.6.8.6 Concurrent Sequences

No changes from FC-PH.

23.6.8.7 N_Port End-to-end Credit

No changes from FC-PH.

23.6.8.8 Open Sequences per Exchange

No changes from FC-PH.

23.6.9 Vendor Version Level

Vendor Version Level is a 16-byte field which specifies Vendor-specific information. If Word 1, bit 29 = 1 in the Common Service Parameters, the Vendor Version Level field contains valid information. If Word 1 bit 29 =0, the Vendor Version Level is not meaningful.

23.7 F_Port Service Parameters

23.7.1 F_Port Common Service Parameters

Figure 67 illustrates F_Port Common Service parameters, enhanced from FC-PH. FC-PH 23.7.1.3 is modified.

23.7.1.3 Common features

- **Word 1, bit 25 Multicast (M)**

  0 = Multicast not supported by the Fabric

  1 = Multicast supported by the Fabric

- **Word 1, bit 24 Broadcast (B)**

  0 = Broadcast not supported by the Fabric

  1 = Broadcast supported by the Fabric

- **Word 1, bit 22 Dedicated Simplex (D)**

  0 = Dedicated Simplex not supported by the N_Port

  1= Dedicated Simplex supported by the N_Port

- **Word 1, bits 27, 26, 23, 21 -16 [c]**

Reserved.

word	31 – 16	15 – 0
0	FC-PH / FC-PH-2 Version HHHHHHHH LLLLLLLL	Buffer-to-buffer Credit BBBBBBBB BBBBBBBB
1	Common Features rrrNrrMB rDrrrrrr	Buffer-to-buffer Rcv Data Field size r r r r F F F F FFFF FFFF
2	R_A_TOV tttttttt tttttttt	tttttttt ttttttt
3	E_D_TOV tttttttt tttttttt	tttttttt tttttttt

Figure 67 – F_Port Common Service Parameters - Fabric Login

[c] See FC-PH-3 for additional bit (word 1, bit 26) to support the new use of E_D_TOV timer resolution.

23.7.4 F_Port Class Service parameters

F_Port Class Service Parameters used during Fabric Login is illustrated in figure 67A.

word	31 … 16	15 … 0
0	Service Options V ISS QDCB EEEEEEEE	Initiator Control Not meaningful
1	Recipient Control Reserved for Fabric use	Receive Data Field Size Not meaningful
2	Concurrent Sequences Not meaningful	N_Port End-to-end Credit Not meaningful
3	CR_TOV tttttttt tttttttt	tttttttt tttttttt

Figure 67A – F_Port Class Service Parameters - Fabric Login

23.7.4.1 Class validity

- **Word 0, Bit 31 - Class validity (V)**

  0 = Invalid (Class not supported)

  1 = Valid (Class supported)

The Class validity bit shall indicate whether this Class is supported or not. If the Class validity bit is zero, it indicates that this group or set of sixteen bytes shall be ignored. If the Class validity bit is one, it indicates that this Class shall be supported. The Class shall be identified based on Class 1 = first sixteen-byte group, Class 2 = second sixteen-byte group, Class 3 = third sixteen-byte group, and Class 4 = fourth sixteen-byte group.

23.7.4.2 Service options

Service options shall specify optional features of a Class of Service supported by the N_Port supplying the Service Parameters.

- **Word 0, bit 30 (I)**

Class 1, 2 and 3

No change from FC–PH.

Class 4

Word 0 bit 30 (Intermix Mode) has no meaning in Class 4 since the Fabric shall support Intermix on Class 4.

- **Word 0, bits 29 and 28 (SS)**

Class 1

Word 0, bits 29 and 28 of the F_Port Class Service Parameters (FC-PH Clause 23.7.4) are used in the Fabric Login procedure. The ability to support the SCR function by a Fabric is indicated by bits 29 and 28 of word 0 of the F_Port service parameters during Fabric Login. The following set of values specifies the meaning of the combination of word 0 bits 29 and 28:

Table 102 – SCR support Login bits

Service Parameter Word 0		Meaning
Bit 29	Bit 28	
0	0	SCR not supported
0	1	Lock-Down Mode
1	0	Transparent Mode
1	1	Invalid

Class 2 and 3

No change from FC–PH.

Class 4

Word 0, bits 29 and 28 (Stacked connect-request - Transparent and Lock-down modes) are reserved.

- **Word 0, bit 27 (Q)**

Class 1, 2 and 3

No change from FC–PH.

Class 4

Word 0, bits 27 (Sequential Delivery)-

Bit 27 has no meaning in Class 4 since the Fabric shall deliver frames in the order transmitted based on class of service.

- **Word 0, Bit 26 Dedicated Simplex (D)**

Class 1

Word 0, bit 26

0=Dedicated Simplex not supported

1=Dedicated Simplex supported

See clause 33.

Class 2, 3 and 4

Word 0, bit 26 is reserved.

- **Word 0, Bit 25 Camp-On (C)**

Word 0, bit 25
0 = Camp-On not supported
1 = Camp-On supported

See clause 35.

Class 2, 3 and 4

Word 0, bit 25 is reserved.

- **Word 0, Bit 24 - Buffered Class 1 (B)**

0 = Buffered Class 1 not requested or Buffered Class 1 is not supported

0 = Buffered Class 1 requested or Buffered Class 1 is supported

See clause 37.

Buffered Class 1

Support for Buffered Class 1 Service is negotiated during Fabric Login. If an N_Port requests Buffered Class 1 Service and the attached F_Port supports it, only then is Buffered Class 1 Service enabled and available for use by the N_Port and Fabric. Table 101A summarizes the meaning of the combination of Word 0, bit 24.

Class 1, 2, 3 and 4

Word 0, bit 24 is reserved.

23.7.4.3 Initiator Control

No changes from FC-PH.

23.7.4.4 Recipient Control

No changes from FC-PH.

23.7.4.5 Receive Data_Field Size

No changes from FC-PH.

23.7.4.6 Concurrent Sequences

No changes from FC-PH.

23.7.4.7 N_Port End-to-end Credit

No changes from FC-PH.

23.7.4.8 Open Sequences per Exchange

No changes from FC-PH.

23.7.4.9 CR_TOV

The CR_TOV value shall be specified as a count of 1 ms increments. Therefore a value of hex ’0000 000A’ specifies a time period of 10 ms. See 36.8.1 for CR_TOV usage. CR_TOV is applicable to Class 1. CR_TOV is not applicable to other Classes.

24 Exchange, Sequence, and sequence count management

No enhancements to FC-PH are specified.

25 Association Header management and usage

No enhancements to FC-PH are specified.

26 Flow control management

Enhancements to FC-PH 26.5 are specified. There are no changes to other sections of FC-PH clause 26.

26.5 Buffer-to-buffer flow control

26.5.10 Alternate buffer-to-buffer Credit management

An alternate buffer-to-buffer credit management may be used instead of the one described in FC-PH 26.5.1 through 26.5.9 (see annex Y for example implementation in FC-AL).

The use of alternate buffer-to-buffer Credit management shall be indicated by the N_Port through an N_Port Login Service Parameter during Fabric Login and N_Port Login (see FC-PH-2 23.6.2.3 and 23.6.3.3).

26.5.11 Alternate BB_Credit management rules summary

Alternate BB_Credit management rules are summarized:

a) Each Port is responsible for managing the Alternate BB_Credit.

b) During Login, BB_Credit shall be set to a value that represents the number of receive buffers that a Port shall guarantee to have available as soon as a circuit is established. If this value is greater than zero, the Port may start transmitting a frame without waiting for R_RDYs. If this value is equal to zero, the sending Port has to wait to receive at least one R_RDY, before transmitting a frame.

c) The receiving Port shall transmit at least one R_RDY , representing the number of additional receive buffers available, before, during, or after the reception of frames.

d) The transmitting Port shall decrement BB_Credit by one for each frame transmitted and increment by one for each R_RDY received.

e) For transmitting frames, the Available Credit shall be greater than zero. The Available Credit at any given time is expressed by the following equation:

Available Credit =

Login_BB_Credit +

(Number of R_RDYs received - Login_BB_Credit)* -

Number of frames transmitted

where

Number of R_RDYs received ≥Login_BB_Credit.

* The parenthetical expression is applicable only if it is positive, otherwise it is zero.

27 Segmentation and reassembly

No enhancements to FC-PH are specified.

28 Connection management

No enhancements to FC-PH are specified.

29 Error detection and recovery

Timer enhancements to FC-PH are specified.

29.2 Timer enhancements

FC-PH defined two timers, E_D_TOV and R_A_TOV, that are used for normal frame flow. In general, E_D_TOV is used by both the Sequence Recipient (SR) and the Sequence Initiator (SI) to time FC2 inter-frame arrival of data and ACK frames respectively to determine if an error has occurred. R_A_TOV is used as the maximum transit time within a Fabric to guarantee that a lost frame will never emerge from the Fabric after an R_A_TOV time-out.

In general, it is desirable to keep these timers small to minimize the interval required to recognize time-out errors. However, under certain conditions, it is desirable that these time-out values be much larger to allow frames to remain outstanding for an extended time without reporting an error. Specifically, connect request frames under Stacked Connect Request (SCR) are required to remain active within the Fabric for an extended period of time to accomplish their intended functions. The time-out value requirement for these connect request frames and all other frames are considerably different.

This section defines a new timer called Connection Request Time-Out Value (CR_TOV) that is used by the Connection Initiator (CI), the Connection Recepient (CR), and the Fabric to control the connect request process when Stacked Connect Request is invoked.

29.2.6 Terms

The following new terms are used within this section:

- CR_TOV Connection Request Time-out value

29.2.7 Applicability

The CR_TOV timer shall be used by the CI, CR, and the Fabric to time connect request frames that are transmitted under Stacked Connect Request.

29.2.8 Login

CR_TOV is specified by the F_Port during Fabric Login as part of F_Port Class Service Parameters (see 23.7.4).

29.2.9 Value

The minimum CR_TOV value shall be RA_TOV. The maximum value of CR_TOV is set by the Fabric.

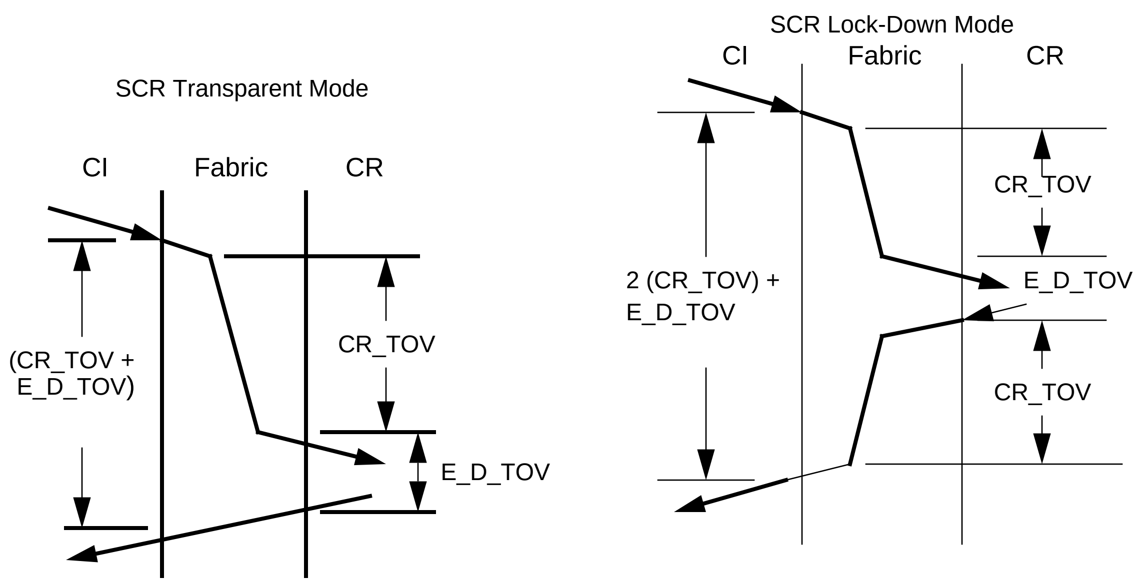


Figure 87 – Frame Flow Timers

29.2.10 Stacked Connect Request

The CR_TOV timer shall be used by the Fabric to define the maximum time period that the Fabric shall hold a Stacked Connect Request frame.

29.2.11 Rules

29.2.11.1 Connection Initiator

– Following the transmission of a connect request frame invoking Stacked Connect Request, if the CI receives an F_BSY in reply to that connect request, the CI must retry that connect request.

– The CI shall wait a minimum time period of

CR_TOV + E_D_TOV

in SCR Transparent Mode following the transmission of a connect request frame invoking Stacked Connect Request before it can declare the connect request frame to be lost and initiate error recovery.

– The CI shall wait a minimum time period of

2 (CR_TOV) + E_D_TOV

in SCR Lock_Down Mode following the transmission of a connect request frame invoking Stacked Connect Request before it can declare the connect request frame to be lost and initiate error recovery.

29.2.11.2 Connection Recipient

– The CR shall wait a minimum time period of

E_D_TOV

following receipt of a connect request frame which invoked Stacked Connect Request Transparent mode, before considering the ACK to the CR to have been lost.

– The CR shall wait a minimum time period of

CR_TOV + E_D_TOV

following receipt of a connect request frame which invoked Stacked Connect Request Lock Down mode, before considering the ACK to the CR to have been lost.

29.2.11.3 Fabric

– The Fabric shall start the CR_TOV timer on receipt of a connect request frame which has invoked Stacked Connect Request.

– If the CR_TOV timer expires on a connect request frame which has invoked either Stacked Connect Request or Camp-On before the Fabric can forward it to the CR, the Fabric shall discard the connect request frame and Fabric transmit an F_BSY frame to the CI.

30 Hunt Group

30.1 Introduction

A Hunt Group is a collection of one or more N_Ports controlled by a Common Controlling Entity. A Hunt Group is addressed using a Hunt Group Identifier (HG_ID) which is managed in FC-PH-2 as an Alias. The Hunt Group shall be formed by the registration of a group of one or more N_Ports with the Alias Server. An N_Port may be a member of multiple Hunt Groups and may be addressed by either its base address identifier or any one of its Hunt Group Identifiers (Aliases). Other methods may be used to select one of N destination N_Ports using only base address identifiers, however, those methods do not constitute a Hunt Group.

N_Port Login with a Hunt Group using the designated HG_ID shall be required prior to other communication. Login with a Hunt Group using the HG_ID as the destination address identifier may be accomplished by an N_Port using its base address identifier or HG_ID (Alias) and the normal N_Port Login procedure. The formation and operation of a Hunt Group shall require specific support in both the N_Port and the Fabric.

The Alias Server handles the registration of N_Ports into a Hunt Group, and the deregistration of N_Ports from a Hunt Group. The Alias Server is not involved in the routing of Hunt Group frames.

Hunt Group routing is part of the adaptive addressing protocol specified in 30.7.

30.2 Function

The function of a Hunt Group is to provide a HG_ID (Alias) which shall be used by the Fabric to select a single available N_Port out of the group of possible N_Ports as the destination (D_ID) of a frame. This increases the likelihood of locating a destination N_Port which is not engaged in a Class 1 connection. In Class 2, Hunt Groups may also be used to increase aggregate bandwidth, reduce latency, or both.

30.3 Communication Model

The Hunt Group communication model is an extension of N_Port to N_Port communication. Since a Hunt Group is a collection of N_Ports, the following combinations are supported in this model:

- N_Port to N_Port
- N_Port to Hunt Group
- Hunt Group to Hunt Group

All combinations are supported as part of the adaptive addressing protocol, specified in 30.7.

30.4 Applicability

Hunt Group is an optional communication model. To support this model, Fabric and N_Port shall have the additional capability specified.

30.4.1 Classes

Class 1, 2, 3, or 4 may be used with Hunt Group. Common Controlling Entity (CCE) based or Port based allocation of resources required shall vary by Class of service (see 30.7.1).

30.4.2 Class 4 Addressing

Class 4 S_ID addressing is different from other Classes due to support of multiple Virtual Circuits per N_Port in Class 4. Since Class 4 S_ID is always associated with a specific Virtual Circuit ID, any Class 4 S_ID used in Hunt Group shall be understood to have a specific Virtual Circuit ID associated with it, explicitly stated or not.

This address implication shall be valid for all mechanisms specified for Hunt Group for Class 4 N_Ports.

For example, an N_Port ID K1 in Class 4 implies specifically, say, K1(5) where 5 is the VIrtual Circuit ID. A Hunt Group ID K in Class 4 implies specifically, say, K1(5), K2(9), and K3(25).

30.5 Formation

A Hunt Group is formed by an N_Port under control of the Common Controlling Entity performing Hunt Group Registration with the Alias Server. The N_Port performing the registration need not be a member of the Hunt Group being formed but shall be under control of the Common Controlling Entity constituting the Hunt Group.

The ultimate responsibility lies with Common Controlling Entity in allowing any of its N_Ports becoming a Hunt Group member and communicating common service parameters for such membership. The method by which the Common Controlling Entity informs its N_Ports is outside the scope of this document.

During registration, the Alias Server shall inform each member N_Port of its HG_ID.

Providing Hunt Group information to a Fibre Channel Directory Server or Management Server is the responsibility of the Alias Server. At the end of the Hunt Group registration, the Alias Server has the following information:

– Alias assigned

– Assigned Alias is associated with Hunt Group

– Node Name associated with the list of N_Ports in the Hunt Group

– List of N_Ports in the Hunt Group.

The Fabric Controller is responsible for distribution of Aliases for Hunt Group Identifiers and Fabric-N_Port Service Parameters to the appropriate internal facilities and F_Ports within the Fabric. The method for distributing this information within the Fabric is outside the scope of this document.

30.5.1 Registration / deregistration

Registration of a list of N_Ports into or deregistration of a list of N_Ports from a Hunt Group by a Common Controlling Entity is performed by originating an Exchange with the Alias Server. Any N_Port under control of the Common Controlling Entity may be the Originator of the Exchange performing Registration / deregistration in a Class supported by the Fabric Controller and the N_Port. The third party authorization for registration or deregistration is the responsibility of the Common Controlling Entity which is outside the scope of the document.

The Exchange may be used to:

– create a Hunt Group with a list of member N_Ports,

– add a list of N_Ports to an existing Hunt Group, or

– delete a list of N_Ports of an existing Hunt Group.

Hunt Group Identifier is managed in FC-PH-2 as an Alias. The detailed format and protocol for registration / deregistration are specified in clause 32.

The information transmitted by the Originator contains the following items:

Node_Name

The Node_Name of the Common Controlling Entity.

HG_ID

The Hunt Group Identifier which is associated with an existing Hunt Group for add and delete functions. In the case of creating a Hunt Group, the HG_ID may be unidentified (binary zeros) or a value requested by the Common Controlling Entity.

N_Ports

During creation of the Hunt Group, all member N_Ports are identified. During add or delete functions, member N_Ports being added or deleted are identified.

Common and Class Service Parameters

During the creation of a Hunt Group, the Fabric N_Port Service Parameters shared by each member N_Port for all three Classes of Service shall be included in the same format and meaning as during the N_Port to Fabric Login procedure (see clause 23 of FC-PH).

30.5.2 Inquiry

The Hunt Group information may be available from the Directory Server, if Alias Server has communicated the information to the Directory Server.

If the membership of a Hunt Group is uncertain, the Common Controlling Entity may replace its members by using appropriate Alias Server functions for an existing HG_ID.

30.5.3 Hunt Group ID Removal

Removal of a Hunt Group by a Common Controlling Entity is performed by originating an Exchange with the Alias Server. Any N_Port under control of the Common Controlling Entity may be the Originator of the Exchange performing the removal of a Hunt Group in a class supported by the Alias Server and the N_Port. The details of the Exchange are specified in the in clause 32.

30.6 Destination N_Port Login

Destination N_Port Login shall be required on an address identifier basis. Therefore, Login shall be performed with the Hunt Group Identifier (HG_ID) prior to communication other than Basic and Extended Link Applications. Login with any instance of the alias address identifier of HG_ID completes Login with all members of the Hunt Group (HG_ID).

For example, HG_ID=K (K1, K2, K3) and HG_ID=M (M1, M2, M3) exist. M shall be required to Login with K before M to K or K to M communication even if K1 has performed Login with M2. Likewise, K3 shall be required to Login with M2 before K3 to M2 communication even if K and M Login has been performed. Destination N_Port Login shall have been completed between the S_ID and D_ID N_Port address identifiers prior to communication of Upper Level Protocols for the first Data frame of the Exchange.

Hunt Group Service parameters

Each member N_Port of a specific Hunt Group shall operate with a single, identical set of Common and Class Service Parameters with the Fabric and destination N_Ports with which it communicates.

30.7 Addressing Protocol

An adaptive address identifier protocol is utilized when Hunt Group Identifiers (HG_ID) are being employed to allow for multiple strategies for managing Exchange and Sequence resources. This adaptive address identifier protocol supports any combination of communication models (see 30.3).

30.7.1 Originator

The Originator of an Exchange may specify its N_Port ID or an HG_ID as S_ID in the first Data frame of the Exchange. The Originator may specify as the Responder's N_Port ID or an HG_ID as D_ID in that frame.

30.7.2 Responder

In the ACK to that first Data frame, the Responder shall address the Originator with the same ID which the Originator identified itself in the first Data frame. However, the Responder may choose to use either its N_Port ID or the Hunt Group ID with which it is addressed by the Originator.

NOTE - If the Responder specifies another HG_ID, the Originator may reject the Exchange with a reason code = protocol error.

30.7.3 Address Resolution

After the first ACK has been specified, the identifiers for both Originator and Responder shall remain the same, provided X_ID invalidation has not occurred during the Exchange. With X_ID invalidation, the Exchange shall resume from step specified in 30.7.1.

The address resolution for different cases is summarized in the Table 152.

30.7.4 Class

The address resolution is applicable to Class 1, 2 and 4, since ACK is used in the protocol. As there are no ACKs in Class 3, address resolution is not possible and HG_ID shall be used throughout the Exchange.

Table 152– Address Resolution

Originator First Data frame	**Responder ACK to First Data frame**	**Resolution**
S_ID, D_ID	**S_ID, D_ID**	**Originator ID, Responder ID**
Originator HG_ID K, Originator N_Port ID x	Originator N_Port ID x, Originator HG_ID K	Originator HG_ID K, Originator N_Port ID x
Responder N_Port ID y, Responder HG_ID L	Responder HG_ID L, Responder N_Port ID y	Responder N_Port ID y, Responder HG_ID L
	Responder N_Port ID L_m, Responder N_Port ID y	Responder N_Port ID y, Responder N_Port ID L_m
Originator HG_ID K, Responder HG_ID L	Responder HG_ID L, Originator HG_ID K	Originator HG_ID K, Responder HG_ID L
Originator N_Port ID Kn, Responder HG_ID L	Responder N_Port ID L_m, Originator N_Port ID K_n	Originator N_Port ID K_n, Responder N_Port IDL_m
	Responder HG_ID L, Originator N_Port ID K_n	Originator N_Port ID K_n, Responder HG_ID L

Legend:
x = Originator N_Port ID y = Responder N_Port ID
K_n = Originator N_Port ID L_m = Responder N_Port ID
K = Originator HG_ID L = Responder HG_ID

30.7.5 Class 4

The address resolution summarized in table 152 shall be valid for Class 4 also, with the Class 4 address implication specified in 30.4.2.

30.8 Resource management

Since a Hunt Group may span multiple N_Ports, resources may be managed by the Common Controlling Entity and by N_Port. The sharing of resources may be implemented in various combinations. Some possible implementable combinations are included in annex X. The arrangement of control and status block elements are only logical and does not necessarily imply any physical relationship.

An N_Port uses S_ID, D_ID, X_ID (OX_ID for Originator, RX_ID for Responder), SEQ_ID and Context bits to uniquely identify frames within a Sequence received by the N_Port. An N_Port is allowed to use any or all of the control entities for a given implementation.

When an N_Port supports multiple address identifiers or common address identifiers for S_ID, D_ID, or both, Sequence and frame identification alters N_Port behavior. The manner in which Hunt Group Identifiers are utilized within an N_Port leads to various logical models of N_Port resources (see annex X).

Based on the organization of resources within an N_Port and CCE and the Class of Service, the Originator or Responder shall choose HG_ID or N_Port ID for an Exchange.

30.8.1 HG_ID use

For an Originator or a Responder to use a HG_ID, the following resource requirements shall be met:

- Class 1
  - Exchange resources (ESB, OX_ID) shall be Common Control Entity (CCE) managed
  - Sequence resources (SSB) may be N_Port managed or CCE managed
- Class 2
  - Exchange resources (ESB, OX_ID) shall be CCE managed
  - Sequence resources (SSB) shall be CCE managed
- Class 3
  - Exchange resources (ESB, OX_ID) shall be CCE managed
  - Sequence resources (SSB) shall be CCE managed
- Class 4
  - Exchange resources (ESB, OX_ID) shall be CCE managed
  - Sequence resources (SSB) shall be CCE managed

30.8.2 N_Port ID use

For an Originator or a Responder to use its N_Port ID, there are no additional requirements for resource organization. All resources may be CCE managed or N_Port managed or shared between them. If shared, the SSB, ESB, X_ID control hierarchy shall be retained.

30.9 Rotary Group

Rotary Group is another approach to increase the bandwidth and reduce latency but does not fall under the category of Hunt Group.

Existence of Rotary Group shall be transparent to compliant N_Ports or F_Ports.

30.9.1 Introduction

A Rotary Group is a collection of one or more N_Ports controlled by a Common Controlling Entity and a cooperative Fabric (See FC-SW - Rotary Groups) in which any member of the Rotary Group may respond in a transparent way to non-participating N_Ports. In this way, systems which can handle more bandwidth than the attached N_ports provide can utilize multiple N_Port to Fabric connections to extend their achievable bandwidth.

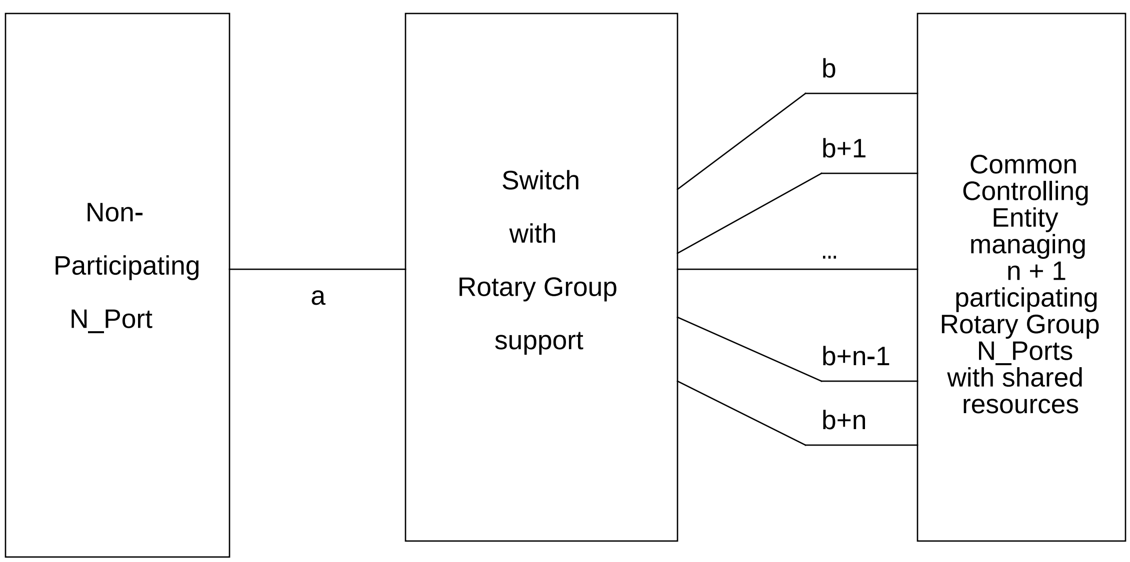
Rotary Group are formed and managed on a local basis, both in the participating N_Ports host (Common Controlling Entity) and in the Fabric, by the System administration.

The overall objective of a Rotary Group is that other N_Ports or Directory Services interact with the Group in a completely transparent manner (i.e. N_Ports which are not members of the Group do not know or care about the existence or non existence of the Rotary Group). Non-participating N_Ports address any N_Port of the Rotary group by its native address. No matter which N_Port of the Rotary Group responds, the Non-Participating N_Port receives acknowledgments and other communications associated with an established Exchanges with the Group as if it were communicating only with the native address of the Group with which it had originally established the exchange.

30.9.2 Function

In Class 1 a Switch With Rotary Group support may be configured administratively to connect any connection request for a specific member of a Rotary group to any available member of that group. In Class 2 and 3 a Switch with Rotary Group support may be configured administratively to route any frame addressed for a specific member of a Rotary group to any available member of that group. (See FC-SW - Rotary Groups)

A Rotary Group member N_Port, under direction of its Common Controlling Entity,



Legend: (a = address of the Non Participating N_Port, b through b+n = addresses of n Participating N_Ports with one Common Controlling Entity)

Figure 88 –Relationship of a Non-Participating to a Participating N_Port and switch

returns the address with which it was addressed as it source address (no matter which Rotary N_Port is actually comes from) in instances where it is the exchange responder. The Common Controlling Entity may use any available N_Port of its Rotary Group that is logged in to a destination N_Port for those exchanges where it is the exchange Originator. The Common Controlling Entity manages the resources (i.e. buffers, state tables, etc.) as a pool for all of the participating Rotary Group N_Ports.

31 Multicast

31.1 Introduction

Multicast provides an unacknowledged multicast service based on Fabric routing of Class 3 frames. When the Fabric receives a Class 3 frame that is to be multicast, it causes the frame to be delivered to every destination in the Multicast Group. A Multicast Group is a list of N_Ports to which a frame being multicast is delivered. A Multicast Group is managed by the Alias Server, which handles the registration of N_Ports into a Multicast Group, and the deregistration of N_Ports from a Multicast Group. A Multicast Group is identified with an MG_ID. The Alias Server is not involved in the routing of multicast frames.

Multicast routing is specified in 31.3 in more detail.

31.2 Registration and Deregistration

The registration/deregistration by an N_Port as a member of a Multicast Group with the Alias Server is specified in clause 32. An N_Port may register itself as a member of one or more Multicast Groups. An N_Port may register a list of N_Ports as members of one or more Multicast Groups. The third party authorization for registration or deregistration is outside the scope of the document.

31.3 Multicast Routing

All routing of multicast frames is done by the Fabric, based on a recognition that the D_ID of the transmitted frame is an MG_ID. The exact frame that entered the Fabric is replicated to every destination N_Port in the Multicast Group associated with the MG_ID of the frame. The Fabric shall not alter the frame header or the frame contents in any manner during this replication.

The Sequence Initiator performs no special function to transmit a frame to be multicast other than use a D_ID indicating the MG_ID and the Multicast Group Service Parameters, rather than the Login Service Parameters. For example, the Receive Data Field Size for a multicast frame may be different than the Receive Data Field Size for a unicast frame.

If the Sequence Recipient is a member of a Multicast Group, it shall recognize the MG_ID as an Alias and accept the frame.

Class 1, Class 2, and Class 4 frames with a D_ID equal to an MG_ID shall be rejected by the Fabric with an F_RJT containing the reason code = "protocol error".

Figure 89 illustrates the multicast routing. N_Port B, C, D, and E are members of a Multicast Group. N_Port A is the message sender which may not be a member of the group. If it is a member, it may optionally receive the same message (see FC-GS).

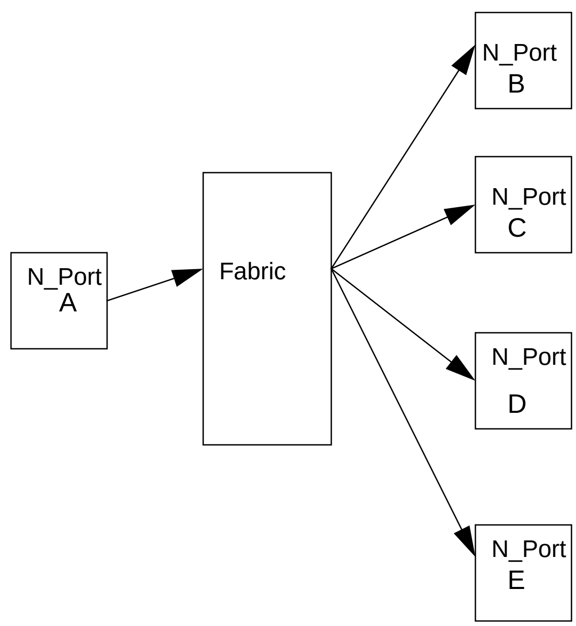


Figure 89 –Multicast Routing

31.4 Multicast rules

a. The Sequence Initiator of the Multicast message shall follow the uniqueness rules for OX_ID, SEQ_ID, and Sequence Count. RX_ID for the Sequence shall be set to hex'FFFF'.
b. D_ID in all frames shall be the MG_ID for the Multicast Group. S_ID in all frames shall be the N_Port Identifier of the Sequence Initiator.
c. The Fabric shall not alter the frame in any way.The Fabric shall simply replicate each frame to every group member.
d. Sequence Initiative shall not be transfered.

e. If any group member is not powered on nor able to receive the frame for any other reason, the Fabric shall not retransmit the frames.
f. All multicast frames shall be routed in Class 3. Class 1, Class 2, and Class 4 frames with a D_ID equal to an MG_ID shall be rejected by the Fabric with an F_RJT containing the reason code = "protocol error".

31.5 Broadcast

Broadcast is considered a simplification of multicast. No explicit registration or de-registration is required. Hex ‘FFFFFF’ is the Well-known address for Broadcast which may be recognized by an N_Port.

The Broadcast Service Parameters shall be the default service parameters before Login (see FC-PH 21.1.1).

31.6 Moviecast

Moviecast is an inherent feature of Multicast. For example, if a source N_Port is multicasting a movie to an Alias Group, another N_Port may join this Alias Group “in progress”. That is, after the registration procedure is complete, the Fabric will start routing frames to the new N_Port.

31.7 Other

Other forms of Multicast are available in topology specific configurations. For example, see FC-AL for a description of Selective Replicate to perform dynamic multicasting.

32 Aliases

32.1 Introduction

Alias is a group address recognized by an N_Port if the N_Port has registered with the Alias Server as a member of the group. Thus an N_Port may be able to recognize one or more Aliases in addition to its N_Port Identifier.

Presently defined Aliases are:

a. Hunt Group Identifier
b. Multicast Group Identifier.

A given N_Port may register with one or more Hunt Groups and/or with one or more Multicast Groups.

NOTE - Well-known addresses specified in Table 33 are allocated for special functions such as Directory Server. These special addresses are not to be confused with Aliases.

32.2 Alias Server

The Alias Server manages the registration and deregistration of Alias IDs for both Hunt Groups and Multicast Groups. The Alias Server is not involved in the routing of frames for any Group.

The Alias Server may be internal or external to the Fabric, but, in either case, it is addressed by means of the well-known address identifier, hex'FFFFF8'. The following sections describe the registration/ deregistration process in more detail.

Authorization for Alias Server operations is provided.

32.3 Alias Service protocol

Alias registration and deregistration are managed through protocols containing a set of request/reply IUs supported by the Alias Server. These requests and replies use FC-PH constructs as defined in FC-GS.

32.4 Alias Routing

All routing of frames is done by the Fabric, based on a recognition that the D_ID of the transmitted frame is an Alias_ID. For Multicast groups, the exact frame that entered the Fabric is replicated to every destination N_Port in the Multicast Group associated with the Alias_ID of the frame. The Fabric shall not alter the frame header or the frame contents in any manner during this replication. For Hunt Groups, the exact frame that entered the Fabric is routed to a single destination N_Port in the Hunt Group.

NOTE – The Fabric may assign Alias_IDs to easily partition Multicast Group Alias_IDs from Hunt Group Alias_IDs.

The Sequence Initiator performs no special function to transmit a frame other than to use a D_ID indicating the Alias_ID and to use the Alias Group Service Parameters, rather than the Login Service Parameters. For example, the Receive Data Field Size for a multicast frame may be different than the Receive Data Field Size for a unicast frame.

If the Sequence Recipient is a member of an Alias Group, it shall recognize the Alias_ID as an alias address identifier and accept the frame.

For Multicast Groups, Class 1 and Class 2 frames with a D_ID equal to an Alias_ID shall be rejected by the Fabric.

32.5 Function Flow

Figure 90 illustrates the flow among the Originator N_Port, participating N_Ports, Alias Server, Directory Server, and Fabric Controller to create an Alias Group

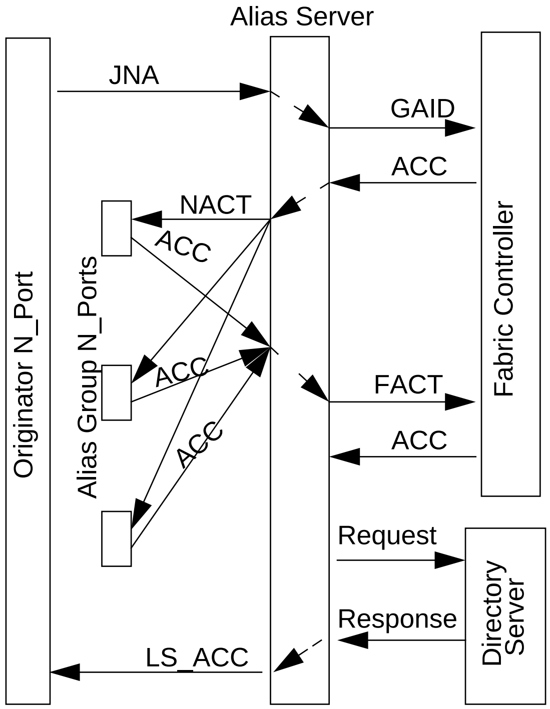


Figure 90 –Function Flow

32.6 IPA Considerations

32.6.1 Hunt Groups

For Hunt Groups, there are no IPA considerations, since there is a requirement that all members of a Hunt Group be within a single Common Controlling Entity, which cannot be spread across images. Therefore, the same IPA may be used no matter which N_Port receives the frame.

32.6.2 Multicast Groups

For Multicast Groups, the considerations for multicasting to N_Ports that require an Initial Process Associator are minimal as any multicasting behind the N_Port is handled internally. It is not necessary to know the IPA of each image behind an N_Port that belongs to a Multicast Group. Instead, if the Multicast Group requires an IPA, then a Multicast Group IPA (MG_IPA) is used by the Sequence Initiator. This MG_IPA is common for all images that belong to the same Multicast Group. An MG_IPA is set in the Association Header as follows:

- The Responder Process Associator is set equal to the Alias_Qualifier for the Multicast Group.
- Bit 24 of the Association_Header Validity bits is set to binary ‘1’. This indicates an MG_IPA.

Internally, images shall register with their N_Ports to join Multicast Groups, they do not register with the Alias Server. The MG_IPA becomes an alias for the actual IPA of the image and is recognized as such by the N_Port as a result of the internal registration. When a frame is received, the N_Port “multicasts” the payload to all images in the internal Multicast Group, based on the MG_IPA.

32.6.3 Broadcast

For Broadcast, there are no IPA considerations. Since the intent of Broadcast is to deliver to all possible recipients, it is the responsibility of the N_Port receiving a broadcast frame to broadcast the payload internally to all its images. Therefore, an IPA shall not be included in a frame being Broadcast.

33 Dedicated Simplex

33.1 Introduction

The Class 1 Dedicated Connection of FC-PH is a duplex, connection-oriented communication service. FC-PH-2 extends this dedicated connection service to include a Dedicated Simplex communication service. Thus, Class 1 service has two variants, duplex and simplex. This clause specifies Class 1 Dedicated Simplex.

Dedicated Simplex provides a means to combine the advantages of both the existing Class 1 and Class 2 in a single communications model as shown in figure 91. It provides the high bandwidth flow-through of Class 1 Data frames for large block transfers with the ability of uncoupling the inbound and outbound data path (fibres) of Class 2. Dedicated Simplex allows a connection to exist on the outbound fibre to one N_Port while simultaneously having a connection on the inbound fibre with another N_Port. It increases the likelihood of simultaneous flow of data over both the outbound and the inbound fibre of a N_Port when compared with Class 1 dedicated connections.

33.2 Terms

The following terms are used with regard to Dedicated Simplex:

– Dedicated Simplex

– Class Specific Control (CS_CTL) field of Frame Header

– Simplex bit of CS_CTL field

33.3 Communications Model

The Dedicated Simplex communications model is essentially a unidirectional connection as defined in FC-PH terms with ACK's transmitted in Class 2. More specifically, a unidirectional dedicated connection is established between the outbound fibre of the Connection_Initiator (CI) and the inbound fibre of the Connection_Recipient (CR). This is shown as paths 'M' and 'N' in Figure 91 and as paths '1' and '3' in figure 92. Path '1' ('M') is a Dedicated Simplex connection from A (CI) to B (CR)and path '3' ('N') is a second (independent) Dedicated Simplex connection from C (CI) to A (CR). The in-bound fibres of the CI's and the out-bound fibres of the CR's are independent of the Dedicated Simplex connection. The CR's (destination N_Ports) will transmit all acknowledgments in Class 2 over paths 'm' and 'n' in figure 91 and paths '2' and '4' in figure 92.

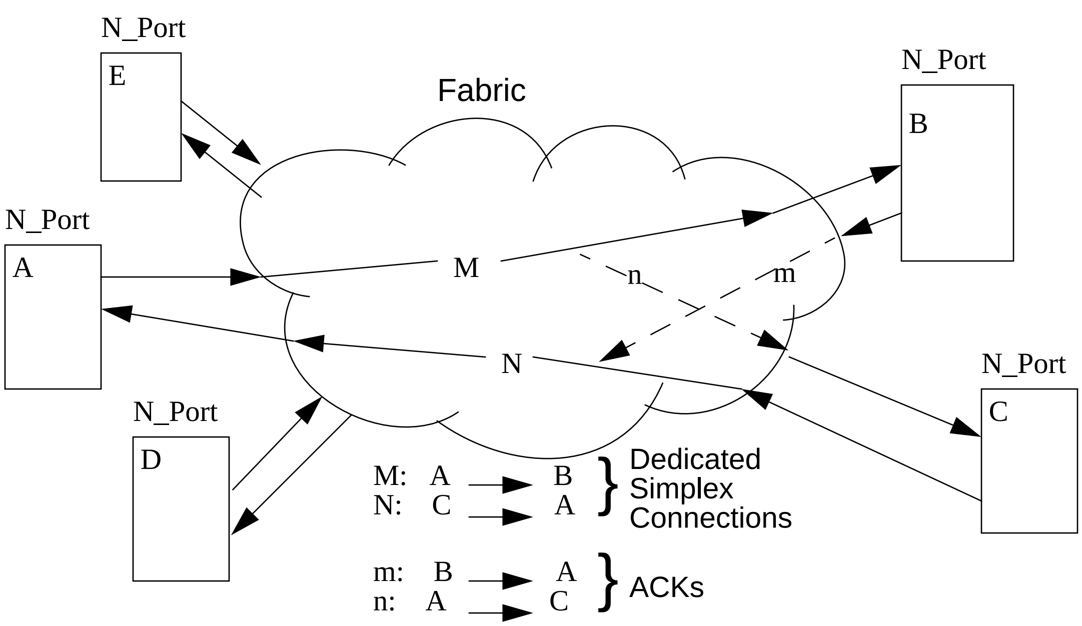


Figure 91 – Dedicated SImplex

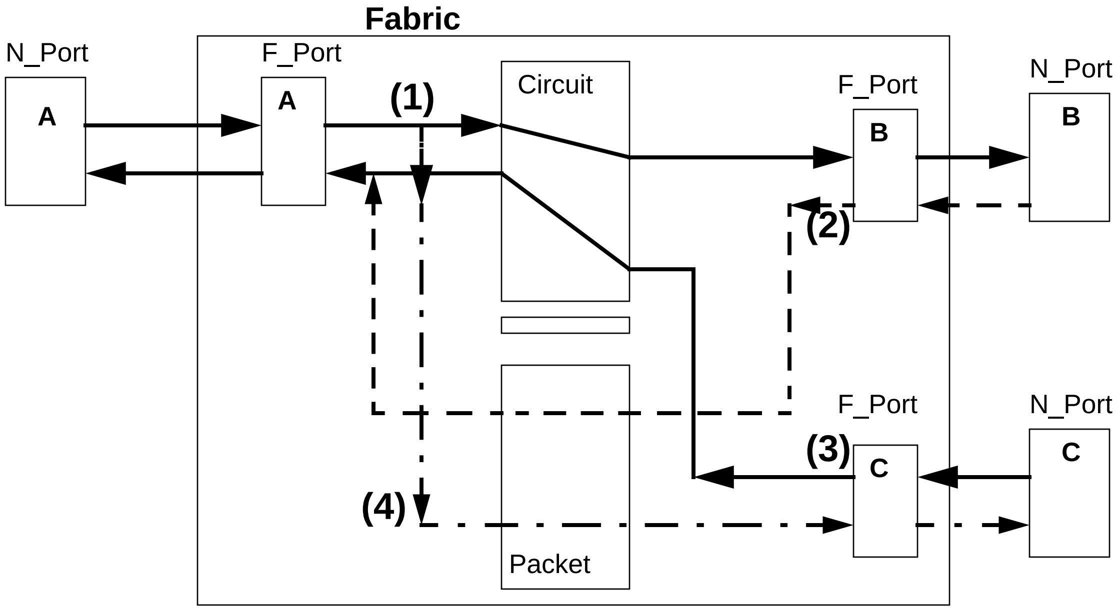


Figure 92 – Dedicated Simplex Fabric Circuit and Paths

Within a Dedicated Simplex connection, the CI shall transmit data frames in sequential order to the Fabric. The Fabric, if present, shall guarantee delivery of data frames within a Dedicated Simplex connection to the CR in the same order as they were received from the CI. The ACK's may be transmitted by the CR in any order and the Fabric, if present, may re-order them prior to delivery to the CI. Thus, Dedicated Simplex maintains in order delivery of data frames, while permitting the potential for out of order delivery of ACK's.

Dedicated Simplex is indicated by the CI through the use of a control bit in the frame header CS_CTL field (see 18.2) of the connect request frame containing the SOFc1 delimiter.

A Dedicated Simplex connection is removed by the CI by transmitting an EOF(dt) delimiter on the last data frame (E_S=1) of the last sequence for this connection. The CR will respond by transmitting a Class 2 ACK with a EOFt delimiter containing the E_S=1.

When the Fabric receives a frame with the SOFc1 delimiter, it will check the Simplex bit of the CS_CTL field of the frame header and take the following action.

– If the Simplex=0, the Fabric shall attempt to establish a Dedicated Duplex connection.

– If the Simplex=1, the Fabric shall attempt to establish a Dedicated Simplex connection.

On receipt of a frame containing a EOFdt delimiter and Simplex =1, the Fabric shall remove the Dedicated Simplex connection associated with the link on which the EOFdt was received (CI outbound fibre and the CR inbound fibre).

For Dedicated Simplex connections, the ACK, P_BSY, F_BSY, P_RJT and F_RJT frames are transmitted in Class 2 with EOFn or EOFt delimiters.

33.4 Applicability

Dedicated Simplex is an optional service that may be provided by the Fabric and N_Ports.

NOTE – Best efforts have been made to show applicability differences to FC-PH.

33.4.1 Permitted Functions

Dedicated Simplex may be used in conjunction with many of the functions defined in FC-PH. When FC-PH functions are used, Dedicated Simplex will adhere to the rules defined in FC-PH except where specifically noted. The following list contains some of the FC-PH functions that may be used in conjunction with Dedicated Simplex:

– Sequence and Exchange management

- X_ID Interlock
- X_ID Reassignment
- X_ID Invalidate
- Stacked Connection request
- Camp-On (proposal)
- Intermix mode
- Continue Sequence Conditions
- Abort Sequence Conditions
- Login
- Class 1 End-to-End Credit rules
- Class 1 Receive Data Field Size
- Class 1 connect request data field limits
- SOFc1, SOFi1, SOFn1, SOFn2
- EOFn, EOFt, EOFdt, EOFa, EOFni, EOFdti

33.4.2 Restricted Functions

Dedicated Simplex shall not use the following FC-PH functions:

- SOFi3, SOFn3
- End_Connection bit protocol

33.4.3 Topology

Dedicated Simplex is applicable to the following topologies:

- Point-to-Point
- Fabric
- Arbitrated Loop

33.5 Requirements

33.5.1 Connection_Initiator N_Port

Dedicated Simplex requires that the CI N_Port has the capability to:

- support Class 1 and Class 2 delimiters
- support Intermix mode
- transmit the Class 1 connection request frame with Simplex=1.
- recognize Class 2 ACK (and other Link_Control frames) for a sequence established in Dedicated Simplex

33.5.2 Connection_Recipient N_Port

Dedicated Simplex requires that the CR N_Port has the capability to:

- support Class 1 and Class 2 delimiters
- support Intermix mode
- support Dedicated Simplex connections

33.5.3 Fabric

Dedicated Simplex requires that the Fabric has the capability to:

- support Dedicated Simplex connections
- support Intermix mode
- support Class 1 and Class 2 delimiters

33.6 Connection Management

Class 1 shall use the Simplex bit (bit 31) of the CS_CTL field (word 1 bit 31-24) of the frame header to differentiate between Duplex and Dedicated Simplex (see 18.2.1).

33.7 Login

N_Ports and Fabrics shall indicate their ability to support Dedicated Simplex connections through Login. During Login, there is no additional requirement on any control bit values to support Dedicated Simplex function.

33.7.1 N_Port Class Service Parameters

The N_Port Class1 Service Parameters (see FC-PH 23.6.7) Service Options field shall be used to indicate support for Dedicated Simplex connections (see 23.6.7.2).

33.7.2 F_Port Class Service Parameters

The F_Port Class1 Service Parameters (FC-PH 23.7.4) Service Options field shall be used to indicate support for Dedicated Simplex connections (see 23.7.4.2).

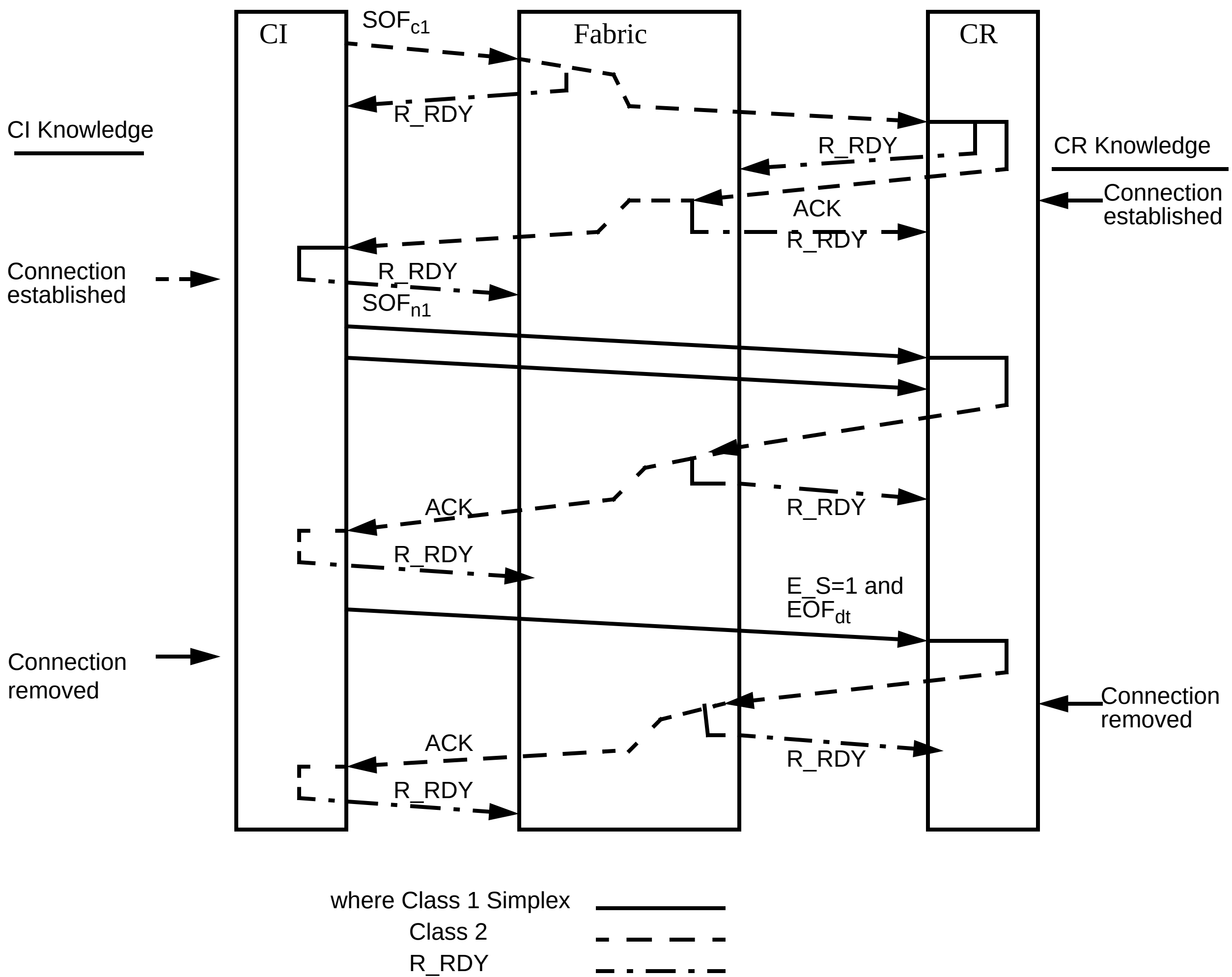


Figure 93 – Dedicated Simplex Frame Flow

33.8 Delimiters

No new delimiters are required for Dedicated Simplex. Use of EOFdt with Dedicated Simplex differs (see 33.3 and 33.9.2) from Dedicated Connection.

33.8.1 Start of Frame

The CI shall use the SOFc1 delimiter together with the CS_CTL Simplex bit to establish a Dedicated Simplex connection. Once the Dedicated Simplex is established, the CI shall use the Class 1 delimiters SOFi1 and SOFn1 as defined in FC-PH for Class 1.

The CR and Fabric shall use the SOFn2 delimiter on Link_Control frames (ACK, P_BSY, F_BSY, P_RJT, and F_RJT) transmitted.

If an N_Port (F_Port) which does not support Dedicated Simplex connections receives a frame with a SOFc1 with the Simplex=1 (bit 31) of the CS_CTL field, that N_Port (F_Port) shall issue a Class 2 P_RJT (F_RJT) frame with a reason code of ‘Dedicated Simplex not supported’ (see table 55).

33.8.2 End of Frame

The CI shall use EOFn, EOFt, and EOFa delimiters as defined in FC-PH. The CI shall also use the EOFdt delimiter to remove a Dedicated Simplex connection.

The CR shall use the EOFn or EOFt delimiter for all Link_Control reply frames transmitted within a Dedicated Simplex connection.

The EOFdti and EOFni rules defined in FC-PH 17.6.2 shall apply to Dedicated Simplex.

33.9 Connections

Dedicated Simplex communication model is a connection-oriented service in one direction combined with connectionless ACK in the opposite direction as shown in figure 93.

33.9.1 Establishment

A Dedicated Simplex connect request is initiated using an SOFc1 delimiter. The CI transmits a connect request frame which contains the SOFc1 delimiter with the Simplex=1 (bit 31) of the CS_CTL field of the frame header. The CI shall follow the rules concerning frame size limits for Class 1 connect request. The CR considers the connection established upon transmission of the Class 2 ACK in reply to the connect request. The CI considers the connection established upon receipt of ACK to its connect request.

33.9.2 Removal

A Dedicated Simplex connection is removed through an EOFdt delimiter. The CI initiates the removal of the connection by placing an EOFdt delimiter on the last frame (E_S=1) of the last sequence of the connection. The CR shall then transmit the last Class 2 ACK with E_S=1 and an EOFt delimiter.

The CI considers the connection to be removed upon transmission of the Data_Frame containing E_S=1 and an EOFdt delimiter. The CR considers the connection to be removed upon receipt of the Class 1 Data_Frame containing the EOFdt delimiter.

If the state of the connection is unknown due to some abnormal condition, the CI may transmit an END Link_Control command (see 20.3.4.3), a NOP or ABTS frame containing a SOFi1, or SOFn1 delimiter (as appropriate) and an EOFdt delimiter. If the state of the connection is unknown to the CR, the CR may transmit a END Link_Control command with an EOFdt delimiter (see 33.11.2).

33.10 Rules

The following rules applies to Dedicated Simplex.

33.10.1 Restriction

The following restriction apply to the Dedicated Simplex class of service:

– A given N_Port shall not be simultaneously involved in a Class 1 Duplex connection and a Class 1 Dedicated Simplex connection. Class 1 Dedicated Duplex and Simplex connections are mutually exclusive. The exclusivity shall be enforced by the F _ Port.

33.10.2 Connection_Initiator N_Port

The following rules apply to the CI:

– The CI shall support transmission of Data_Frames with Class 1 delimiters and Link_Control frames with Class 2 delimiters.

– The CI shall support reception of Link_ Control and Link_Data frame with Class 2 delimiters.

– The N_Port shall support Intermix mode.

– An N_Port shall initiate a Dedicated Simplex connection by transmitting a frame containing the SOFc1 delimiter with the CS_CTL Simplex bit set to a one.

– The CI shall adhere to all the rules of Class 1 connect request frames with the exception that the Simplex=1 shall be used to establish the Dedicated Simplex connection.

– The CI shall consider the Dedicated Simplex connection established upon receipt of the ACK to the connection request frame.

– The CI shall not transmit any frames within the Dedicated Simplex connection following the connection request frame until the ACK frame is received or the appropriate time out period has expired as follows:

-- If Stacked Connect Request Transparent mode has been invoked, the time out period shall be (CR_TOV) + (E_D_ TOV)

-- If Stacked Connect Request Lock-Down mode has been invoked, the time out period shall be 2(CR_TOV) + (E_ D_TOV).

– All frames transmitted by the CI within the Dedicated Simplex connection following the connection request frame shall be transmitted with Class 1 SOFi1 or SOFn1 delimiters.

– Frames transmitted by the CI within a Dedicated Simplex connection shall adhere to the Class 1 credit management rules.

– The Sequence Count rules of FC-PH 24.3.6 shall apply to Dedicated Simplex connections.

– Following transmission of a Class 1 Dedicated Simplex connection request frame, if the CI receives a connect request frame for a Duplex Class 1 prior to the ACK to the connect request and if Stacked Connects has not been invoked, the CI shall re-queue the connect request frame for retransmission at a latter time.

– Following transmission of a Class 1 Dedicated Simplex connect request, if the CI receives a connect request frame for a Class 1 Dedicated Simplex prior to the ACK to the connect request, the CI shall reply with an ACK. From the perspective of this CI, it will be engaged in two Dedicated Simplex connections.

– An N_Port may transmit a Dedicated Simplex connect request with Stacked Connect Request invoked while engaged in a Dedicated Simplex connection which it initiated.

– An N_Port may transmit a Dedicated Simplex connect request with Stacked Connect Request invoked while engaged in a Dedicated Duplex connection.

– The CI shall initiate the removal of a Dedicated Simplex connection by transmitting a frame containing the E_S=1 and an EOFdt delimiter.

33.10.3 Connection_Recipient N_Port

The following rules apply to the CR:

– The CR shall support reception of Data frames with Class 1 delimiters and Link_ Control with Class 2 delimiters.

– The CR shall support transmission of Link_Control and Link_Data frame with Class 2 delimiters.

– The CR may reply to a connection request with either P_BSY or P_RJT transmitted in Class 2.

– The CR completes the establishment of a Dedicated Simplex connection by transmitting an ACK in Class 2 in reply to the connection request frame. Following transmission of the ACK frame in reply to the connection request frame, the CR shall consider the Dedicated Simplex connection to have been established.

– The CR shall respond to all frames received within the Dedicated Simplex connection Class 2 with SOFn2 and EOFn delimiter except a frame with E_S=1. Class 2 shall be used to transmit any Link_Control frames (ACK, P_BSY, or P_RJT) in response to the Connect frame.

– In response to the Data_Frame containing E_S=1 and with an EOFdt delimiter, the CR shall transmit an ACK frame with E_S=1 and an EOFt delimiter.

– The CR shall consider the Dedicated Simplex connection to have been removed upon receipt of a frame from the CI containing the EOFdt delimiter.

– If a CR which does not support Dedicated Simplex receives a Dedicated Simplex connect request frame, that N_Port shall issue a Class 2 P_RJT with reason code of ‘Class not supported’.

33.10.4 Fabric

Fabric shall obey the following rules:

– When the Fabric receives a Dedicated Simplex connect request frame with a destination N_Port which is not engaged in a Class 1 Duplex connection or the inbound fibre of that destination N_Port is not engaged in a Class 1 Dedicated Simplex connection, the Fabric shall attempt to establish a Dedicated Simplex connection with the destination N_Port.

– When the Fabric receives a Dedicated Simplex connect request frame with SCR=1, the Fabric shall follow the Stacked Connection rules as defined in FC-PH 28.5.2 and clause 36.

– When the Fabric receives a Dedicated Simplex connect request frame with COR=1, the Fabric shall follow the Camp-On rules as defined in clause 35.

– The Fabric shall respond to all frames received within the Dedicated Simplex connection in Class 2 with SOFn2 and EOFt delimiters. Class 2 will be used to transmit any Link_Control frames (F_BSY, or F_RJT) in response to the Connect frame.

– Within a Dedicated Simplex connection, if the Fabric receives an END Link_Control command, it shall remove that connection.

– If a Fabric which does not support Dedicated Simplex receives a Dedicated Simplex connect request frame, that F_Port shall issue a Class 2 F_RJT with reason code of 'Class not supported'.

33.10.5 F_Port

F_Port shall obey the following rules:

As specified in FC-PH 28.3.1, the F_Port acts as the control point in resolving collisions or accepting the connect-requests.

– The F_Port shall resolve the collision, if any, between simultaneous connect-requests from the Link side and the internal side. These requests may be for Dedicated Simplex or Dedicated Duplex connections.

-- If these simultaneous requests relate to the same fibre pair or an individual fibre, then a collision results and the request from internal side shall always be accepted by the F_Port and the Link side request shall be discarded.

-- If these simultaneous requests relate to different fibres, then no collision results and the F_Port may be able to accept both the requests.

– After the F_Port has accepted a Dedicated Simplex connect-request from the Link side, only its inbound fibre is considered busy and the outbound fibre is available for Dedicated Simplex connect-request from internal side.

33.10.6 ACK Processing and Routing

Following rules apply to ACK processing and routing:

– With the Dedicated Simplex service, the CR shall transmit ACK's in Class 2.

– Any of the ACK forms (0/1/N) may be used. The History protocol for ACK's as defined in FC-PH 20.3.2.2 may be used.

– The CI may invoke the credit window rules defined in FC-PH 24.3.9, item d) to detect a missing ACKs.

NOTE – The Fabric intermixes ACKs in place of Idles. Intermixing ACKs has an additional Fabric requirement which is different from Intermix defined in FC-PH.

33.11 Error Recovery

33.11.1 Connection_Initiator

The CI performs the following for error recovery:

– The CI may invoke either the ABTS or ABTX protocol to abnormally terminate a Sequence or Exchange on detection of an abnormal condition.

– If the state of the connection is unknown due to some abnormal condition, the CI may transmit a END Link_Control command, NOP, ABTS, or ABTX frame containing a SOFx1 delimiter (as appropriate) and an EOFdt delimiter.

– E_D_TOV and R_A_TOV timer rules apply to frame transmitted within a Dedicated Simplex connection.

33.11.2 Connection_Recipient

The CR performs the following for error recovery:

– If the CR detects an error in the Sequence_Count of the frames received within a Dedicated Simplex connection, it shall invoke the Abort Sequence protocol through use of the Abort Sequence Conditions in the F_CTL of the ACK frame.

– E_D_TOV and R_A_TOV timer rules apply to frame transmitted within a Dedicated Simplex connection.

– If the state of the connection is unknown to the CR, the CR may transmit an END Link_Control command with an EOFdt delimiter.

34 Class 4 – Fractional

34.1 Introduction

Class 4 Service is a connection oriented service for use with a Fabric. This is an optional service allowing the establishment of a Class 4 circuit between a pair of communicating N_Ports. The Class 4 circuit provides a fractional allocation of the resources of the path between the N_Ports.

A Class 4 circuit is bidirectional with one Virtual Circuit (VC) operational in each direction. Each VC may have different Quality of Service (QoS) parameters. These QoS parameters include guaranteed bandwidth and latency. An N_Port may have up to 254 coexistent Class 4 circuits with the same or different N_Ports.

Class 4 operation is separated into two parts, circuit setup and circuit activation (see figure 94). During the setup process, QoS parameters for both VCs are provided by the circuit initiator N_Port (CTI) and granted by the Fabric and the circuit recipient N_Port (CTR). The Fabric establishes a Class 4 circuit, in the Dormant state when the setup is granted by the Fabric. The Class 4 circuit setup process incur a one time round trip delay.

During the circuit activation process, both the Dormant VCs of the Class 4 circuit are put in the Live state to allow communication between the associated CTI and CTR.

The resources of an established Class 4 circuit are available on call so participating N_Ports can activate the circuit and deactivate it repeatedly (n) without paying any round trip delay penalty. Activation of a Class 4 circuit places it in the Live state and deactivation places the Class 4 circuit in the Dormant state.

A QoS Facilitator (QoSF) function is provided within the Fabric to establish and maintain QoS parameters for the Dormant VCs and manage them when the VCs are alive.

The Fabric regulates available bandwidth per Live VC by controlling the outbound frame flow out of the associated N_Ports. This is achieved by controlling credit, between the N_Port transmitter and the F_Port receiver, on a per VC basis, with the F_Port regulating credit to ensure the QoS parameters. This level of control permits congestion management for a Fabric.

Class 4 frames are never busied, since F_BSY and P_BSY are illegal responses to Data and Link_Control frames in Class 4.

In-order delivery of frames within a Class 4 circuit is guaranteed in Class 4 Service.

In Class 4, the Fabric is responsible for multiplexing frame belonging to different Live VCs between the same or different N_Port pairs.

34.1.1 Topology

Class 4 is applicable only to topologies where single N_Ports are attached to F_Ports.

34.2 Communication Model

Class 4 service provides a guaranteed bandwidth between communicating N_Ports. The Extended Link Services associated with Class 4 sets up a pair of unidirectional VCs between communicating N_Ports (see figure 95) to guarantee bandwidth and frame transmission delay.

Although there is a fixed route (path) through the Fabric for each VC between the communicating N_Ports and the Fabric regulates the

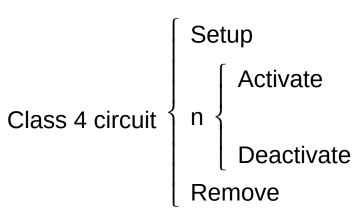


Figure 94 – Class 4 circuit hierarchy

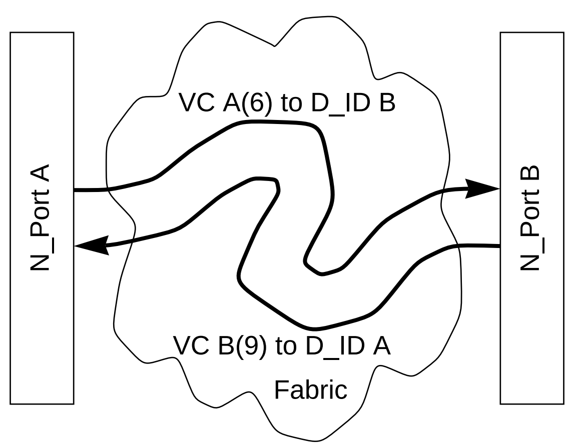


Figure 95 – Basic Class 4 circuit

VC_Credit, the path bandwidth may not be fully dedicated.

The Fabric regulates the N_Ports for frame transmission per VC. However, the communicating N_Ports may be unable to use the full requested bandwidth. This happens if there are no frames to transmit for the VC, if VC_Credit is unavailable or if EE_C4_Credit is unavailable. As a result that amount of bandwidth may be lost to the N_Port. It is possible that this lost bandwidth can be compensated for later, since excess bandwidth is allotted to Live Class 4 circuits and Class 2 or 3 traffic.

Multiple pairs of VCs may exist between one N_Port and one or more other N_Ports (see figure 96). The amount of bandwidth needed for each VC is requested when the Class 4 circuit is setup.

34.2.1 In-order delivery

In-order delivery of Class 4 frames sent, on a Class 4 circuit, is guaranteed. However, no guarantees exist for frames sent on separate Class 4 circuits.

34.2.2 Guaranteed delivery

Class 4 provides acknowledged delivery of frames. This acknowledgment comes in the form of ACK, F_RJT or P_RJT Link_Control frames.

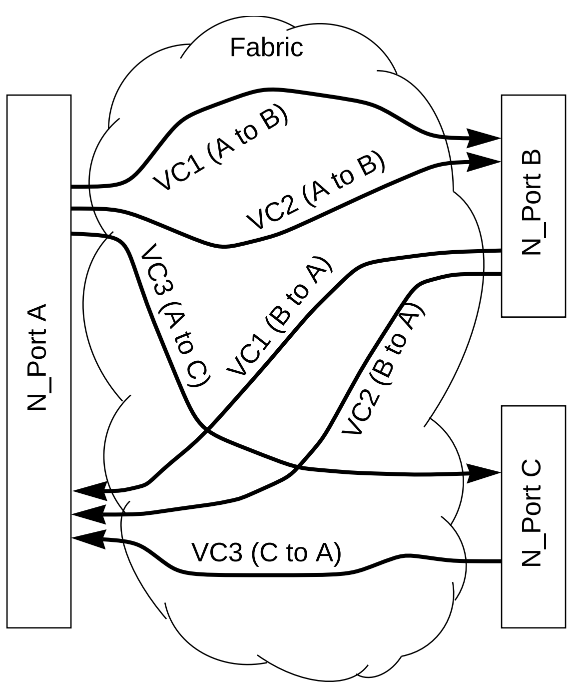


Figure 96 – Class 4 circuits – Example

34.2.3 Fractional bandwidth management

For each Live VC the Fabric ensures the guaranteed bandwidth from a source N_Port to a destination N_Port. A Quality of Service Facilitator (QoSF) manages the guaranteed bandwidth. The Fabric enables a source N_Port to transmit frames in a timely manner.

In Class 4 the source N_Ports transmit frames into the Fabric when credit is available and the Fabric regulates available credit per VC to the source N_Port.

The Fabric routes frames based on the Virtual Circuit Identifier (VC_ID) field embedded in each Class 4 frame header (see figure 96).

The Fabric shall make unused bandwidth available for Live Class 4 circuits, Class 2 and Class 3 frames. Unused bandwidth available for Live Class 4 circuits, shall be allocated approximately in proportion to their requested maximum bandwidth requirements.

The Fabric may ensure that an N_Port does not transmit frames on a VC in excess of the requested maximum bandwidth limit.

34.2.4 Class 4 circuit

A Class 4 circuit requires that two unidirectional VCs be setup through the Fabric before Class 4 frames may be transmitted between the N_ Ports. Figure 95 shows a Class 4 circuit between two N_Ports ‘A’ and ‘B’, one VC (VC A(6)) from port ‘A’ to ‘B’ and one VC (VC B(9)) from port ‘B’ to ‘A’. Each VC has associated with it a set of QoS parameters and a path (see 34.2.4.1).

Table 153 illustrates how a Class 4 circuit can be setup between two N_Ports ‘A’ and ‘B’. N_ Port ‘A’ supports a data rate of 100 MB/s, and ‘B’ a data rate of 25 MB/s. The VC from ‘A’ to ‘B’ (VC(1)) has a guaranteed minimum bandwidth of 15 MB/s and a maximum latency of 500 µs. The VC from ‘B’ to ‘A’ (VC(6)) has a guaranteed minimum bandwidth of 3 MB/s and a maximum latency of 2 000 µs. N_Port ‘A’ is left with an excess outbound bandwidth of 85 MB/s and an excess inbound bandwidth of 97 MB/s. The excess bandwidth for N_Port ‘B’ is 22 MB/s in the outbound direction and 10 MB/s in the inbound direction. Excess bandwidth may

Table 153– Bandwidth Allocation Example

Class 4 Connection		N_Port		Rate (MB/s)	Max. Delay (μs)
		ID	VC_ID		
	A ⇒ B	A	(1)	15	500
	A ⇐ B	B	(6)	3	2,000
Free	A ⇒	A	n/a	85	n/a
Free	A ⇐		n/a	97	
Free	B ⇒	B	n/a	22	n/a
Free	B ⇐		n/a	10	

be used to enable transfers of more than the guaranteed minimum on the Class 4 circuit. The excess may be used for other Class 4 circuits or may be used for Class 2 or 3 Service.

Class 4 circuits and their VCs exist in one of four states:

- Nonexisting,
- Pending,
- Dormant, or
- Live.

A Ports state diagram for a Class 4 circuit is shown in figure 97. The states and state transitions are described in the sub-clauses that follow.

Ports that support Class 4 shall, regardless of the state of Class 4 circuits, support Intermix of Class 2 and Class 3 frames based on the rules of those two Classes of service.

34.2.4.1 Class 4 circuit – Setup

An N_Port requests setup of a Class 4 circuit to another N_Port by sending a Quality of Service request (QoSR) Extended Link Service command to the QoSF. The requesting N_Port becomes the circuit initiator (CTI) and the target N_Port the circuit recipient (CTR). The QoSR identifies the communicating N_Ports and the required QoS parameters.

The QoSR is granted if the QoSF can establish a path between the communicating N_Ports with the requested QoS parameters and if the CTR also can accept the QoS parameters (see 21.18.1). If the QoSR is accepted then a Class 4 circuit is established in the Pending state when the ACC Sequence is returned to the CTI (see figure 98 (a)).

The QoSR is rejected if the QoSF is unable to establish a path between the communicating N_Ports with the requested QoS parameters. The CTR is never queried if the QoSF is unable to establish a path between the communicating N_Ports (see figure 98 (b)).

The QoSR is rejected if the QoSF can establish a path between the communicating N_Ports with the requested QoS parameters but the CTR is unable or unwilling to accept the QoSR (see figure 98 (c)).

The CTI selects the QoS parameters for both VCs in the Class 4 circuit.

The QoSF receives the request from the CTI (S_ID = A and D_ID = hex ‘FFFFF9'), validates it, sets up the path from the CTI to the CTR placing the Class 4 circuit in the Dormant state. It then sends a new request with the QoSF as the S_ID to the CTR (S_ID = hex ‘FFFFF9' and D_ID = B).

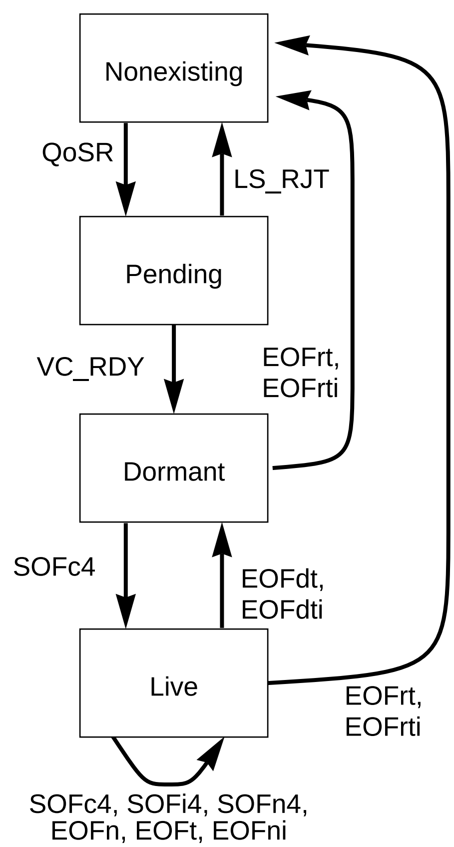


Figure 97 – Class 4 circuit – Port level state diagram

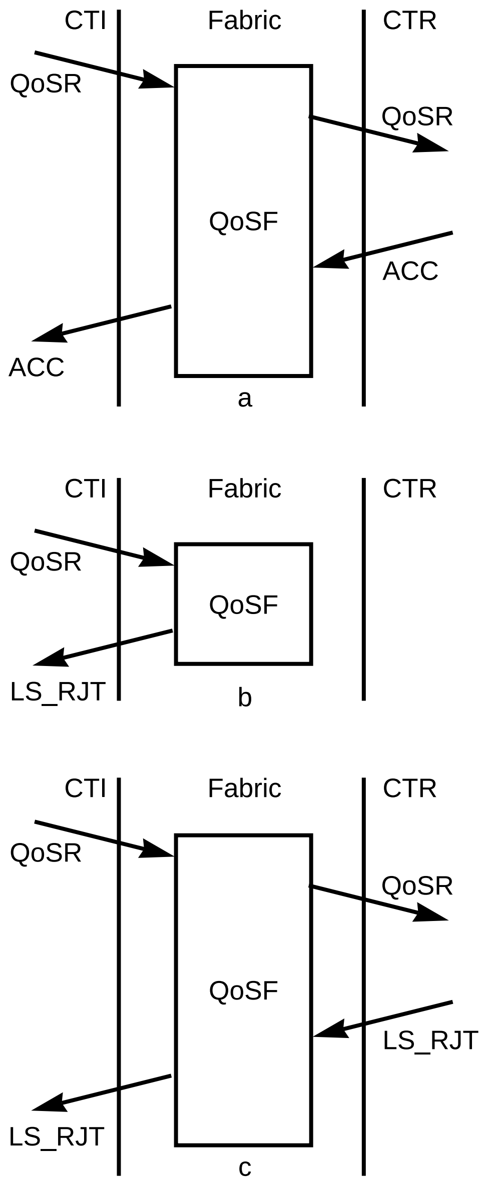


Figure 98 – Class 4 circuit – Setup

If there is a conflict in parameters or insufficient resources, the request shall be rejected. If there is any error in the original transaction, the QoSF shall reject the request (see figure 98 (b)).

General conditions associated with a Class 4 circuit are:

– An N_Port may establish up to 254 coexistent Class 4 circuits, to the same or different N_Ports. The total bandwidth allocated shall not exceed the total bandwidth available for the N_Port media rate of the port managing the Class 4 circuits.

– The bandwidth of any Class 4 circuit is restricted to the maximum bandwidth of the slowest link in the Class 4 circuit path.

– Once setup by the Fabric, a Class 4 circuit is retained and guaranteed by the Fabric. This Fabric service guarantees the requested minimum bandwidth between the communicating N_Ports once the Class 4 circuit is activated.

– The Quality of Service Request (QoSR) Extended Link Service shall be used by a CTI to provide its VC_ID (CIVC_ID) and identify the CTR along with QoS parameters.

The CTI actions are:

– The CTI shall set the VC_Credit_Limit, VC_Credit, EE_C4_Credit_Limit, and EE_C4_Credit to 0 before it sends the QoSR to the QoSF. The CTI shall enter the Pending state for the VC as it sends the QoSR to the QoSF.

– The CTI shall release all resources assigned to the Class 4 circuit if an LS_RJT is received from the QoSF.

– The CTI shall, when it recognizes the ACC Sequence to the QoSR, set the Live_VC_Credit_Limit to the value obtained from the ACC Sequence. The Originating N_Port shall also set the EE_C4_Credit_Limit and the EE_C4_Credit to the value obtained from the ACC Sequence. CTI VC_Credit remains at 0 until an VC_RDY is received on the new Class 4 circuit.

– The CTI shall use the address identifier in the CTR address identifier field in the ACC Sequence, as the D_ID for all Class 4 traffic on the new Class 4 circuit.

NOTE – The CTR may elect to use any address identifier, alias or native, assigned to the CTR on the Class 4 circuit.

– The CTI shall enter the Dormant state when it receives the first VC_RDY for the VC. It shall also set the VC_Credit_Limit to 1 and VC_Credit to 1 when the VC_RDY is received.

The CTR actions are:

– The CTR shall enter the Pending state

before it transmits an ACC Sequence to the QoSR from the QoSF. Before the CTR transmits an ACC Sequence shall the CTR:

- set the VC_Credit_Limit to 0;
- set the Live_VC_Credit_Limit to the value obtained from the QoSR Sequence;
- set the EE_C4_Credit_Limit to the value obtained from the QoSR Sequence;
- set the EE_C4_Credit to the value obtained from the QoSR Sequence; and
- set VC_Credit to 0.

– If the CTR is unable to accept the QoSR from the QoSF it shall return an LS_RJT to the QoSF and enter the Nonexisting state.

– The CTR shall use the address identifier in the CTI address identifier field in the QoSR Sequence, as the D_ID for all Class 4 traffic on the new Class 4 circuit.

NOTE – The CTI may elect to use any address identifier, alias or native, assigned to the CTI on the Class 4 circuit.

– The CTR shall enter the Dormant state when it receives the first VC_RDY for the VC. It shall also set the VC_Credit_Limit to 1 and VC_Credit to 1 when the VC_RDY is received.

The QoSF actions are:

– The QoSF shall set the F_Port, linked to the CTI, to the Pending state, once it recognizes the QoSR from the CTI.

– The QoSF shall set the F_Port, linked to the CTI, to the Dormant state when it completes the QoSR transaction with the CTI. The QoSR transaction is complete in Class 1,2 or 4 when the ACK to the ACC Sequence, sent to the CTI, is received by the QoSF. In Class 3 the QoSR transaction is complete when the ACC Sequence is transmitted by the QoSF. The F_Port shall transmit one or more VC_RDY when the F_Port enters the Dormant state.

– The QoSF shall set the F_Port, linked to the CTR, to the Pending state, as it sends the QoSR to the CTR.

– The QoSF shall set the F_Port, linked to the CTR, to the Dormant state when it completes the QoSR transaction with the CTI. The QoSR transaction is complete in Class 1,2 or 4 when the ACK to the ACC Sequence, received from the CTR, is transmitted by the QoSF. In Class 3 the QoSR transaction is complete when the ACC Sequence is transmitted by the CTR. The F_Port, linked to the CTR, shall transmit one or more VC_RDY when the F_Port enters the Dormant state.

– The QoSF shall send the CTI an LS_RJT if an LS_RJT is received from the CTR.

– The QoSF shall set the F_Port, linked to the CTR, to the Nonexisting state, once it recognize the LS_RJT from the CTR.

– The QoSF shall send the CTI an LS_RJT if it is unable to accept the CIs QoSR request.

– The QoSF shall set the F_Port, linked to the CTI, to the Nonexisting state, before it sends an LS_RJT to the CTI.

– The QoSF shall send ACC to the CTI only if an ACC has been received from the CTR.

– The QoSF shall remove the Class 4 circuit (see 34.2.4.4) if the QoSR transaction fails to complete.

The Class 4 circuit may be activated once it is setup (see 34.2.4.1).

34.2.4.2 Class 4 circuit – Activation

A Class 4 circuit activation request is used to activate the two VCs for the Class 4 circuit after which Class 4 frames may be transmitted between the communicating N_Ports. Either N_port may activate the Class 4 circuit. Once activated, the Class 4 circuit may be deactivated with a deactivation request or removed entirely with a removal request by either of the communicating N_Port.

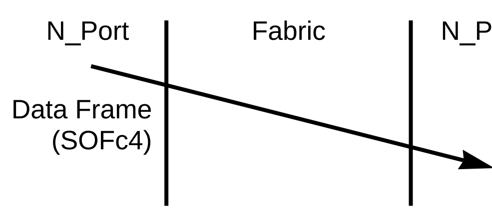


Figure 99 – Class 4 circuit – Activation

A Class 4 circuit, in the Dormant state, is activated, enters the Live state, when a circuit activation request, an SOFc4 delimited frame, is sent or received (see figure 99).

Bandwidth utilization is being monitored by the Fabric and is used to regulate VC_RDY transmission. Once the Class 4 circuit activation frame has been transmitted Class 4 frame flow is possible, only limited by available EE_C4_Credit and VC_Credit.

The Fabric manages buffer-to-buffer flow control and the Fabric regulates frame transmission from each source N_Port. The Fabric regulates each VC independently with the Virtual Circuit Ready (VC_RDY) Primitive Signal. Fabric flow control is accomplished using the VC_RDY Primitive Signal for each Live Class 4 circuit (see figure 100).

Class 4 end-to-end flow control is managed by the communicating N_Ports using ACK frames.

34.2.4.3 Class 4 circuit – Deactivation

Once a Class 4 circuit is setup, it may be activated and deactivated one or more times.

The Sequence Initiator indicates in the frame header, on the last frame of a Sequence, that the Live Class 4 circuit is to be deactivated (E_C = 1, R_C = 0) (see 34.3). The ACK response then carries the EOFdt delimiter (see figure 101).

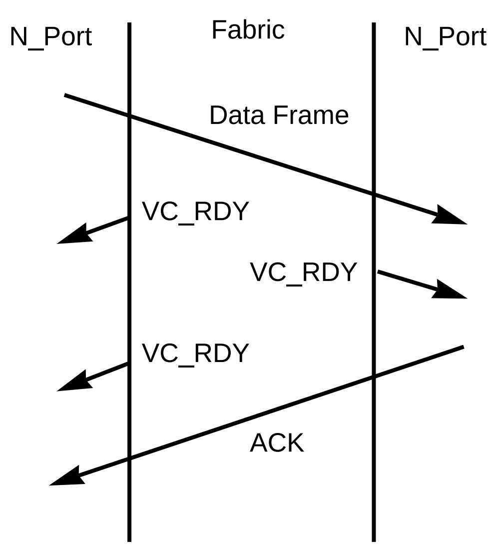


Figure 100 – Class 4 – Example of frame flow

The request to deactivate a Class 4 circuit may be sent by either of the communicating N_Ports.

A Class 4 circuit, in the Live state, is deactivated, and enters the Dormant state, when a circuit deactivation request, an EOFdt or EOFdti delimited frame, is sent or received (see figure 101).

A deactivated Class 4 circuit may later be reactivated to the Live state (see 34.2.4.2).

The Fabric shall issue two END Link_Control commands, one on either VC, when the QoSF determines for any reason that a Class 4 circuit should be deactivated. The Fabric shall use the addresses of the communicating N_Ports, as the S_ID and D_ID, in the generated END Link_Control commands, and the VC_ID shall match with the S_ID field. The END Link_Control command frame delimiters shall be EOFdt and SOFn4 (see figure 102). The Fabric shall use this mechanism to deactivate Class 4 cir-

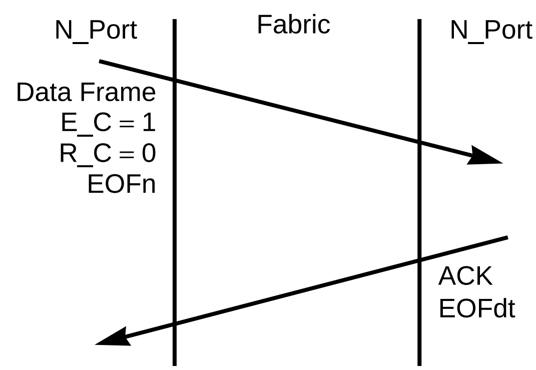


Figure 101 – Class 4 circuit – CTI **or** CTR **initiated**

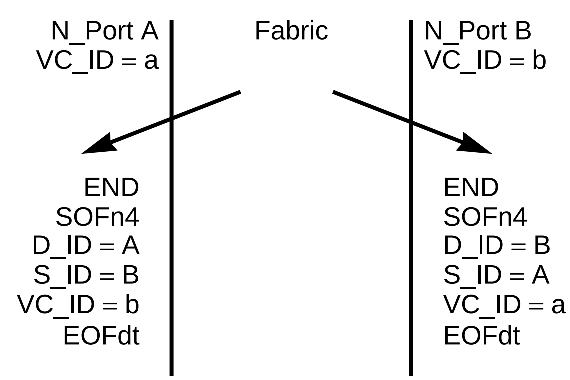


Figure 102 – Class 4 circuit – Fabric initiated deactivation

cuits when the N_Port or F_Port enters the Link Reset state.

When a Class 4 circuit is deactivated, an N_Port shall reclaim all EE_C4_Credit when the N_Port enters the Dormant state (i.e., set EE_C4_Credit equal to EE_C4_Credit_Limit when using the count down model for credit management).

34.2.4.4 Class 4 circuit – Removal

From the Live state, a Class 4 circuit may be removed, i.e. enter the Nonexisting state. The Sequence initiator indicates, in the frame header, on the last frame of a Sequence, that the Live Class 4 circuit is to be removed (E_C = 1, R_C = 1) (see 34.3). The ACK response then carries EOFrt delimiter (see figure 103).

The request to remove a Live Class 4 circuit may be sent by either of the communicating N_Ports.

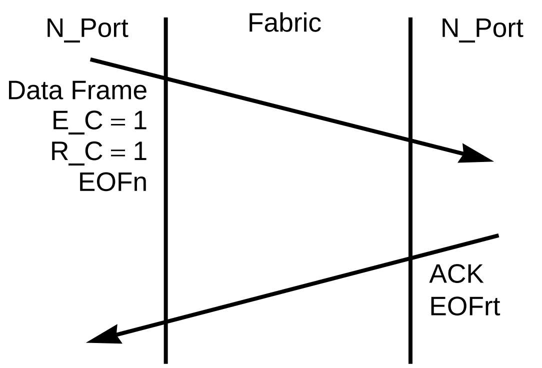


Figure 103 – Class 4 circuit – CTI **or** CTR **initiated removal**

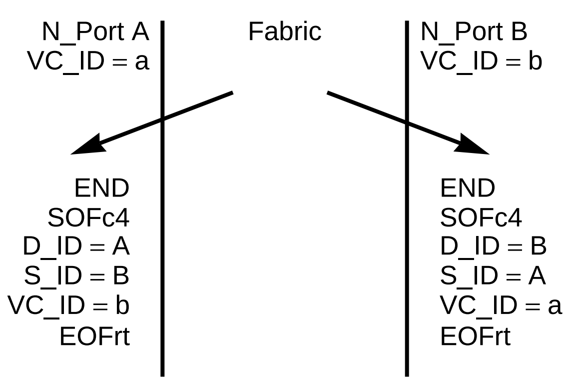


Figure 104 – Class 4 circuit – Fabric initiated removal

A Class 4 circuit, in the Live or Dormant state, is removed, and enters the Nonexisting state, when a circuit removal request, an EOFrt or EOFrti delimited frame, is received (see figure 102). A Class 4 circuit, in the Live state, is removed, and enters the Nonexisting state, when a circuit removal request, an EOFrt or EOFrti delimited frame, is sent (see figure 102).

The two N_Ports shall not reactivate the Class 4 circuit when the Class 4 circuit has been removed. However, the VC_IDs associated with the removed Class 4 circuit (CIVC_ID and CRVC_ID) may be reused. The QoSF shall return any resources it has allocated to a common pool for use by other Live Class 4 circuits or other Classes of Service when the Class 4 circuit has been removed.

Another Class 4 circuit may be used for future Class 4 communication between the N_Ports or the two N_Ports may setup a new Class 4 circuit.

If required for any reason, a Class 4 circuit may be removed by the Fabric from either the Live state or the Dormant state. To achieve this the Fabric shall issue two END Link_Control commands, one on either VC. The Fabric shall use the addresses of the communicating N_Ports, as the S_ID and D_ID, in the generated END Link_Control commands, and the VC_ID shall match with the S_ID field. The END Link_Control command frame delimiters shall be EOFrt and SOFn4 (see figure 104). If the Class 4 circuit is alive, then the Fabric shall first deactivate the Class 4 circuit before removing the circuit (see 34.2.4.3).

The Fabric shall use this mechanism to remove Class 4 circuits when:

- Fabric re-Login is attempted,
- N_Port or the F_Port enters the Offline state,
- Link failure is detected at the N_Port,
- error recovery for the setup process (see 34.2.4.1), or
- internal Fabric failures of the Class 4 circuit path are detected (i.e., the Fabric is unable to provide the guaranteed quality of service).

An N_Port shall reclaim all EE_C4_Credit when the N_Port enters the Nonexisting state (i.e., set the EE_C4_Credit to the EE_C4_Credit_ Limit limit when using the count down credit management model).

34.3 Class 4 circuit – Deactivation, removal contention and error recovery

If a CTI has requested removal of a Class 4 circuit (i.e. set E_C = 1, R_C = 1 on the last Data frame transmitted) and this Data frame has not yet been acknowledged by the CTR, the CTI shall react as follows:

a) The CTI shall acknowledge Data frames received form the CTR on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be either EOFn or EOFt, preventing the CTI from responding to deactivation or removal requests from the CTR.

b) Requests to remove or deactivate the Class 4 circuit from the CTR shall be denied (see 34.3(a)).

c) Data frame rejection (P_RJT) criteria for the CTI is unaffected by the Class 4 circuit removal request.

d) The CTI shall not transmit any Data frames on the Class 4 circuit until either an acknowledgment to the removal request frame has been received or until Sequence timeout is detected.

e) If Sequence timeout is detected, then the CTI may attempt Sequence recovery as specified in FC-PH or it shall transmit an END Link_Control command to deactivate the Class 4 circuit, followed by an END Link_ Control command to remove the circuit.

If a CTR has requested removal of a Class 4 circuit (i.e. set E_C = 1, R_C = 1 on the last Data frame transmitted) and this Data frame has not yet been acknowledged by the CTI, the CTR shall react as follows:

f) The CTR shall acknowledge Data frames received form the CTI on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be EOFn, EOFt or EOFrt.

g) Requests to deactivate the Class 4 circuit from the CTI shall be denied (see 34.3(f)).

h) Requests to remove the Class 4 circuit from the CTI shall be accepted (see 34.3(f)).

i) Data frame rejection (P_RJT) criteria for the CTR is unaffected by the Class 4 circuit removal request.

j) The CTR shall not transmit any Data frames on the Class 4 circuit until either an acknowledgment to the removal request frame has been received or until Sequence timeout is detected.

k) If Sequence timeout is detected, then the CTR may attempt Sequence recovery as specified in FC-PH or it shall transmit an END Link_Control command to deactivate the Class 4 circuit, followed by an END Link_ Control command to remove the circuit.

If a CTI has requested deactivation of a Class 4 circuit (i.e. set E_C = 1, R_C = 0 on the last Data frame transmitted) and this Data frame has not yet been acknowledged by the CTR, the CTI shall react as follows:

l) The CTI shall acknowledge Data frames received form the CTR on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be either EOFn, EOFt or EOFrt, preventing the CTI to respond to deactivation request from the CTR but allowing the CTI to respond to removal request from the CTR.

m) Requests to deactivate the Class 4 circuit from the CTR shall be denied (see 34.3(l)).

n) Requests to remove the Class 4 circuit from the CTR shall be accepted (see 34.3(l)).

o) Data frame rejection (P_RJT) criteria for the CTI is unaffected by the Class 4 circuit deactivation request.

p) The CTI shall not transmit any Data frames on the Class 4 circuit until either an acknowledgment to the deactivation request frame has been received or until Sequence timeout is detected.

q) If Sequence timeout is detected, then the CTI may attempt Sequence recovery as specified in FC-PH or it may transmit an END Link_Control command to deactivate the Class 4 circuit.

If a CTR has requested deactivation of a Class 4 circuit (i.e. set E_C = 1, R_C = 0 on the last Data frame transmitted) and this Data frame has not yet been acknowledged by the CTI, the CTR shall react as follows:

r) The CTR shall acknowledge Data frames received from the CTI on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be EOFn, EOFt, EOFdt or EOFrt, enabling the CTR to respond to both deactivation and removal requests from the CTI.

s) Requests to deactivate or remove the Class 4 circuit from the CTI shall be accepted (see 34.3(r)).

t) Data frame rejection (P_RJT) criteria for the CTR is unaffected by the Class 4 circuit removal request.

u) The CTR shall not transmit any Data frames on the Class 4 circuit until either an ACK frame to the deactivation request frame has been received or until Sequence timeout is detected.

v) If Sequence timeout is detected, then the CTR may attempt Sequence recovery as specified in FC-PH or it shall transmit an END Link_Control command to deactivate the Class 4 circuit, followed by an END Link_Control command to remove the circuit.

If an CTI has no outstanding deactivation or removal request, the CTI shall react as follows:

w) The CTI shall acknowledge Data frames received from the CTR on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be EOFn, EOFt, EOFdt or EOFrt, enabling the CTI to respond to both deactivation and removal requests from the CTR.

x) Requests to deactivate or remove the Class 4 circuit from the CTR may be accepted (see 34.3(w)).

If a CTR has no outstanding deactivation or removal request, the CTR shall react as follows:

y) The CTR shall acknowledge Data frames received from the CTI on the Class 4 circuit. The EOF delimiter used on the ACK frames shall be EOFn, EOFt, EOFdt or EOFrt, enabling the CTR to respond to both deactivation and removal requests from the CTI.

z) Requests to deactivate or remove the Class 4 circuit from the CTI may be accepted (see 34.3(y)).

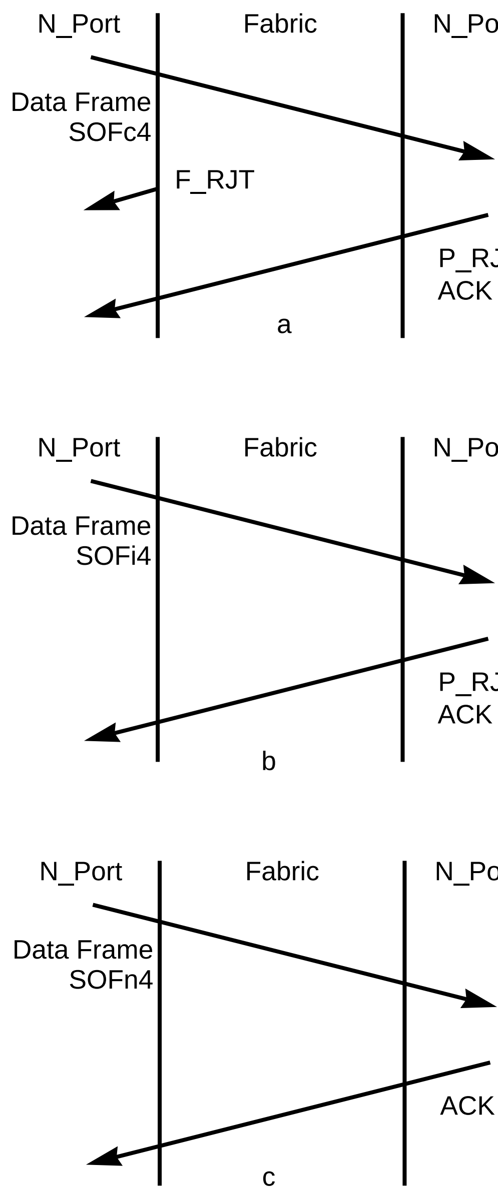


Figure 105 – Class 4 circuit – Data frame responses

Table 154– Frame delimiter usage	
Frame Types	**Delimiters**
Data	SOFc4, SOFi4, SOFn4, EOFn
Link_Control	SOFc4, SOFn4, EOFn, EOFt, EOFdt, EOFrt

34.4 Ordered Sets for Class 4

Class 4 Frame delimiter usage for Data and Link_ Control frames are shown in table 154.

34.4.1 Start Of Frame (SOF)

A Class 4 circuit in the Dormant state is activated to Live state by using an SOFc4 delimiter for the first frame of a Sequence.

The frames that initiate a Sequence while the Class 4 circuit is in the Live state shall begin with an SOFi4 delimiter.

All other frames for a Class 4 circuit in the Live state shall begin with an SOFn4 delimiter.

34.4.1.1 SOF – Circuit Activate Class 4 (SOFc4)

The SOFc4 shall be used to activate a Class 4 circuit. This delimiter shall also identify the first frame of a Sequence. Interaction with the Fabric and the destination N_Port are shown in figure 105 (a).

An N_Port shall only transmit an SOFc4 delimited frame if the Class 4 circuit is in the Dormant state. N_Ports shall discard SOFc4 delimited frames received for Class 4 circuits in the Non-existing or Pending state. Any SOFc4 delimited frames received by an N_Port for a Class 4 circuit in the Dormant or Live state shall qualify for further processing by the N_Port.

The END link_Control command is the only Link_Control frame which shall make use of the SOFc4 delimiter.

Figure 105 (a) shows all possible responses to an SOFc4 delimited data frame, in the absence of errors.

34.4.1.2 SOF – Initiate Class 4 (SOFi4)

The SOFi4 shall be used on the first frame of a Sequence for Class 4 Service, except when SOFc4 is used.

If the Class 4 circuit is not in the Live state the Port shall discard any SOFi4 delimited frame received.

Figure 105 (b) shows all possible responses to an SOFi4 delimited data frame, in the absence of errors.

34.4.1.3 SOF – Normal (SOFn4)

The SOFn4 shall be used for all frames except the first frame of a Sequence for Class 4.

If the Class 4 circuit is not in the Live state the Port shall discard any SOFn4 delimited frame received.

Figure 105 (c) shows all possible responses to an SOFn4 delimited data frame, in the absence of errors.

34.4.2 End_Of_Frame (EOF)

All frames other than the last frame of a Sequence shall be terminated with an EOFn delimiter.

Each Sequence other than a Sequence that deactivates or removes a Class 4 circuit shall terminate with an EOFt delimiter.

The EOFni delimiter shall be used by the Fabric to replace either an EOFn or an EOFt on frames for which the Fabric detects a frame error.

The EOFa delimiter shall be used to terminate a partial frame due to a malfunction during transmission. It is also used by the Fabric to replace missing EOF delimiters or to truncate overlength frames.

34.4.2.1 EOF – Deactivate Terminate (EOFdt)

The EOFdt delimiter shall deactivate the Class 4 circuit.

The EOFdt shall identify the last ACK frame of a Sequence or may be sent on an END Link_ Control frame. It shall indicate that all Class 4 Sequences associated with this Class 4 circuit are terminated. Open Class 4 Sequences for this Class 4 circuit, other than the SEQ_ID specified in the ACK frame containing EOFrt, shall be abnormally terminated and may require Sequence recovery on a ULP protocol-dependent basis.

34.4.2.2 EOF – Deactivate Terminate Invalid (EOFdti)

The EOFdti delimiter shall replace a recognized EOFdt delimiter on a frame with invalid frame content.

All other actions for an EOFdti shall be the same as for an EOFdt delimiter (see 34.4.2.1).

34.4.2.3 EOF – Remove Terminate (EOFrt)

The EOFrt delimiter shall remove the Class 4 circuit.

The EOFrt shall identify the last ACK frame of a Sequence or may be sent on an END Link_Control frame. It shall indicate that all Class 4 Sequences associated with this Class 4 circuit are terminated. Open Class 4 Sequences for this Class 4 circuit, other than the SEQ_ID specified in the ACK frame containing EOFrt, shall be abnormally terminated and may require Sequence recovery on a ULP protocol-dependent basis.

34.4.2.4 EOF – Remove Terminate Invalid (EOFrti)

The EOFrti delimiter shall replace a recognized EOFrt delimiter on a frame with invalid frame content.

All other actions for an EOFrti shall be the same as for an EOFrt delimiter (see 34.4.2.3).

34.4.3 Virtual Circuit Ready (VC_RDY)

The VC_RDY Primitive Signal shall only be transmitted from an F_Port. The VC_RDY shall indicate that a single Class 4 frame is needed from the N_Port if it wish to maintain the requested bandwidth.

VC_RDY transmission shall obey the following:

– A minimum of two Idle Primitive Signals shall separate a VC_RDY from an EOF or SOF delimiter.

– A minimum of two Idle Primitive Signals shall separate a VC_RDY from a R_RDY Primitive Signal.

– At least one Idle Primitive Signal shall separate VC_RDYs.

– The VC_RDY Primitive Signal shall not be transmitted by an N_Port.

An F_Port shall discard any VC_RDY Primitive Signals received.

34.5 Primitive Sequences

The effect of Primitive Sequences on Class 4 circuits depend on the Port state and is described in the following sub-clauses.

34.5.1 LR Transmit state (LR1)

An N_Port in the LR Transmit state:

– shall retain Class 4 circuit states,

– shall set VC_Credit = 0 for all Class 4 circuits,

– may discard Class 4 frames received on Live Class 4 circuits.

The locally attached F_Port in the LR Transmit state:

– shall retain Class 4 circuit states,

– shall discard Class 4 frames destined for the N_Port,

– may discard Class 4 frames received from the N_Port.

The Fabric with the locally attached F_Port in the LR Transmit state:

– shall, for Live Class 4 circuits, send END Link_Control commands to deactivate the Class 4 circuits when the LR Transmit state is entered (see 34.2.4.3),

– may suspend, to remote N_Ports, VC_RDY pacing for all Live Class 4 circuits.

34.5.2 LR Receive state (LR2)

An N_Port in the LR Receive state:

– shall retain Class 4 circuit states,

– shall set VC_Credit = 0 for all Class 4 circuits.

The locally attached F_Port in the LR Receive state:

– shall retain Class 4 circuit states,

– shall discard Class 4 frames destined for the N_Port.

The Fabric with the locally attached F_Port in the LR Receive state:

– shall suspend VC_RDY transmission to the N_Port,

– shall, for Live Class 4 circuits, send END Link_Control commands to deactivate the Class 4 circuits when the LR Receive state is entered (see 34.2.4.3),

– may suspend, to remote N_Ports, VC_RDY pacing for all Live Class 4 circuits.

34.5.3 LRR Receive state (LR3)

An N_Port in the LRR Receive state:

– shall retain Class 4 circuit states.

The locally attached F_Port in the LRR Receive state:

– shall retain Class 4 circuit states,

– shall discard Class 4 frames destined for the N_Port.

The Fabric with the locally attached F_Port in the LRR Receive state:

– shall retain the suspension of VC_RDY transmission to the N_Port,

– shall if suspended retain suspension, to remote N_Ports, of VC_RDY pacing for all Live Class 4 circuits.

34.5.4 NOS Receive state (LF1)

An N_Port in the NOS Receive state:

– shall set all Class 4 circuits to the Nonexisting state.

The locally attached F_Port in the NOS Receive state:

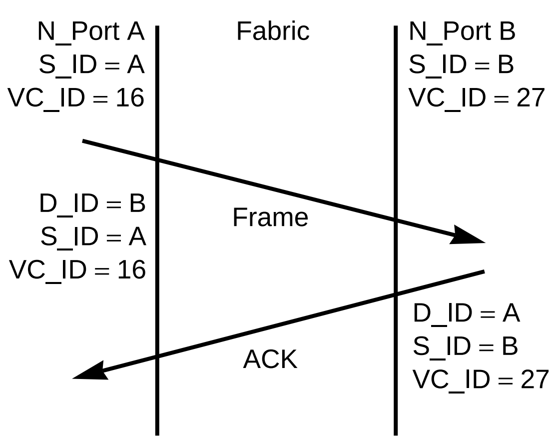


Figure 106 – CIVC_ID and CRVC_ID management

– shall set all Class 4 circuits to the Nonexisting state.

The Fabric with the locally attached F_Port in the NOS Receive state:

– shall remove Dormant or Live Class 4 circuits when the NOS Receive state is entered (see 34.2.4.4).

34.5.5 NOS Transmit state (LF2)

An N_Port in the NOS Transmit state:

– shall set all Class 4 circuits to the Nonexisting state.

The locally attached F_Port in the NOS Transmit state:

– shall set all Class 4 circuits to the Nonexisting state.

The Fabric with the locally attached F_Port in the NOS Transmit state:

– shall remove Dormant or Live Class 4 circuits when the NOS Transmit state is entered (see 34.2.4.4).

34.5.6 OLS Transmit state (OL1)

An N_Port in the OLS Transmit state:

– shall set all Class 4 circuits to the Nonexisting state.

The locally attached F_Port in the OLS Transmit state:

– shall set all Class 4 circuits to the Nonexisting state.

The Fabric with the locally attached F_Port in the OLS Transmit state:

– shall remove Dormant or Live Class 4 circuits when the OLS Transmit state is entered (see 34.2.4.4).

34.5.7 OLS Receive state (OL2)

An N_Port in the OLS Receive state:

– shall set all Class 4 circuits to the Nonexisting state.

The locally attached F_Port in the OLS Receive state:

– shall set all Class 4 circuits to the Nonexisting state.

The Fabric with the locally attached F_Port in the OLS Receive state:

– shall remove Dormant or Live Class 4 circuits when the OLS Receive state is entered (see 34.2.4.4).

34.5.8 Wait for OLS state (OL3)

An N_Port in the ‘Wait for OLS’ state:

– shall retain Class 4 circuits in the Nonexisting state.

The locally attached F_Port in the ‘Wait for OLS’ state:

– shall retain Class 4 circuits in the Nonexisting state.

34.6 Data frames and responses

34.6.1 Class 4 frame size

The maximum frame size used for all frames for a Class 4 circuit shall be the VC Data Field size decided during the QoSR Extended Link Service request.

The Fabric may discard or truncate any Class 4 frames exceeding the Data Field Size limit. If the Fabric truncates a Class 4 frame it shall be terminated with an EOFa delimiter.

34.6.2 QoSF Well-known Address identifier

The well-known address identifier of the Quality of Service Facilitator (QoSF) shall be hex ‘FFFFF9'.

34.6.3 Class 4 – CS_CTL field usage

The Class 4 VC_ID field is a Class of Service specific use of the CS_CTL field of the FC-PH-2 frame header. The VC_ID identifies which VC the frame is associated with from the point of view of the S_ID in the frame header.

An unique VC is identified from S_ID || VC_ID in the frame header. During the Class 4 circuit setup process, the QoSF associates S_ID || VC_ID with a unique destination N_Port D_ID value which is also identified in the QoSR. For each D_ID, the S_ID||VC_ID identifies the route for the VC through the Fabric. The association between VC_ID and S_ID shall be done for the CTI when the QoSR is recognized by the QoSF and for the CTR when it's ACC is recognized by the QoSF.

VC_ID values are assigned as follows:

– The value hex ‘00' is reserved.

– The value in the VC_ID associated with a reserved VC shall be value in the range of hex ‘01' through hex ‘FE'.

– The value hex ‘FF' is reserved.

This assignment of VC_ID values permits a maximum of 254 Dormant or Live Class 4 circuits for each N_Port, shared between the N_Ports address identifier and any alias address identifier.

The Fabric uses the VC_ID field in the frame header to route frames to a destination N_Port. The VC_ID field value is picked by the N_Port during the Class 4 circuit setup process. Each N_Port pick its own VC_ID value.

The value in the field is either a CTI VC_ID (CIVC_ID) or a CTR VC_ID (CRVC_ID), depending on which of the two communicating N_Ports is transmitting the frame.

34.6.4 Frame control (F_CTL)

The E_C bit shall be set to one in the last Data frame of a Sequence to indicate that the N_Port transmitting E_C is beginning the deactivation or removal procedure. The N_Port transmitting E_C = 1 on the last Data frame of a Sequence is requesting the receiving N_Port to transmit an ACK frame terminated by EOFdt or EOFrt if the receiving N_Port has completed all active Sequences. If the R_C = 1 then a Class 4 circuit removal is requested otherwise a deactivation is requested.

If the receiving N_Port is not able to deactivate or remove the Class 4 circuit, E_C = 1 requests that the receiving N_Port complete all active Sequences and not initiate any new Sequences during the current activation cycle.

34.6.5 Data Frames

Permission to transmit frames is granted by a VC_RDY. Each N_Port has a VC_ID that identifies the VC for its half of the Class 4 circuit.

If a data frame is transmitted in a Class 4circuit, the VC_ID field shall contain the appropriate CIVC_ID or CRVC_ID value corresponding to the S_ID field in the same frame (see figure 106).

Table 155– N_Port – Default Class 4 service parameters

Service Parameter	Default value
Class validity	1 (Class supported)
Initiator control	
X_ID reassignment	00 (no reassignment)
Initial CTR Process Associator	00 (not supported)
ACK_0 capable	0 (ACK_0 incapable)
ACK_N capable	0 (ACK_N incapable)
ACK generation assistance	0 (no ACK generation assistance)
Data compression capable	0 (compression incapable)
Data compression History buffer size	00 (History buffer not available)
Recipient control	
ACK_0 capable	0 (ACK_0 incapable)
ACK_N capable	0 (ACK_N incapable)
X_ID interlock	1 (interlock required)
Error Policy supported	00 (discard policy only)
Categories per Sequence	1
Data compression capable	0 (compression incapable)
Data compression History buffer size	00 (History buffer not available)
Receive Data Field size	256
Concurrent Sequences	1
N_Port EE_C4_ Credit	n/a
Open Sequences per Exchange	1

As shown in figure 106, in an ACK frame, the VC_ID field in the frame header is substituted with the VC_ID associated with the switched S_ ID in the frame header (i.e., changed to the other VC_ID associated with the Class 4 circuit).

34.6.6 Link_Continue function

The Link_Continue functions used for Class 4 are

- VC_RDY Primitive Signal (see 34.6.6.1)
- ACK frames (see 34.6.6.2).

34.6.6.1 VC_RDY Primitive Signal

In Class 4, once a Class 4 circuit is activated to provide fractional bandwidth service between communicating N_Ports, the Fabric shall transmit the VC_RDY Primitive Signal to indicate that the receiving N_Port may send a frame for the VC_ID identified in the VC_RDY Primitive Signal.

The VC_RDY Primitive Signal shall only be used for flow control and shall indicate that the VC identified in the VC_RDY still exists in the Fabric. The VC_RDY primitive signal from an F_Port also indicates that a frame buffer is free for the identified VC only.

The VC_RDY Primitive Signal is used for F_ Port-to-N_Port buffer-to-buffer flow control for an individual VC. The VC_RDY Primitive Signal shall be transmitted when the Fabric determines that the N_Port may send a single frame to maintain its QoS parameters for the associated VC. This Primitive Signal shall only be transmitted when a Dormant or Live Class 4 circuit exists with an N_Port.

The VC_RDY Primitive Signal shall indicate that the Source N_Port needs to transmit a single frame, but not to exceed its available EE_ C4_Credit or the VC_Credit, for the indicated VC Identifier (VC_ID).

The event associated with the receipt of a VC_ RDY may be indicated to level above FC-2 within an N_Port to cause a frame for the indicated VC to be transmitted.

34.6.6.2 ACK frames

Class 4 shall use either ACK_1, ACK_0 or ACK_N frames for frame acknowledgment.

The frame’s VC_ID field shall contain the appropriate CIVC_ID or CRVC_ID value. This value may be different from the value in the VC_ID field in the data frame being acknowledged.

The ACK frame used to terminate a Sequence shall have an EOFt delimiter. The ACK frame used to deactivate a Class 4 circuit and terminate a Sequence shall use the EOFdt delimiter. The ACK frame used to remove a Class 4 circuit and terminate a Sequence shall use the EOFrt delimiter. All other ACK frames that do not terminate a Sequence shall use an EOFn delimiter.

34.6.7 Link_Response frames

Class 4 shall use the following Link_Response frames in the same manner as used by Class 1.

- F_RJT (Fabric Reject), or
- P_RJT (N_Port Reject).

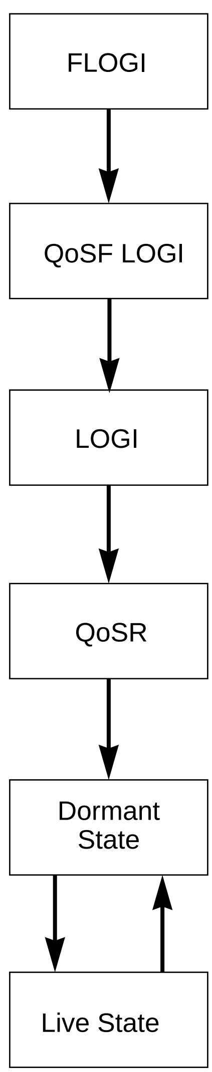


Figure 107 – Class 4 circuit – Management

34.6.7.1 Reject (P_RJT, F_RJT)

Class 4 shall use the F_RJT and P_RJT as specified in FC-PH. Reject Reason Codes specifically for Class 4 are specified in table 91.

If the EE_C4_Credit rules are violated by a source N_Port, the destination N_Port shall reject the frame (P_RJT) with a Reject Reason Code of Protocol Error (see FC-PH, 20.3.3.3).

The frame's VC_ID field shall contain the appropriate CIVC_ID or CRVC_ID value. This value may be different from the value in the VC_ID field in the data frame being rejected.

The Fabric shall not reject any SOFn4 or SOFi4 delimited frames.

The VC_ID field shall be = 0 in an F_RJT frame with the reason code 'Invalid VC_ID'.

Neither the N_Port nor the Fabric shall reject an END Link_Control command. Both the Fabric and the N_Ports shall discard END Link_Control commands with invalid D_ID || S_ID || VC_ID frame header content.

34.6.7.2 Fabric Busy (F_BSY)

The Fabric shall not F_BSY any Class 4 frames.

34.6.7.3 N_Port Busy (P_BSY)

The N_Port shall not P_BSY any Class 4 frames.

34.6.8 Link Credit Reset (LCR) Link_Control command

Class 4 behavior shall not be affected by LCR.

34.6.9 END Link_Control command

The END frame (see 20.3.4.3) shall, in Class 4, indicate that the Class 4 circuit is to be deactivated or removed. The Class 4 circuit is removed if the End Of Frame delimiter is EOFrt or EOFrti. The Class 4 circuit is deactivated if the End Of Frame delimiter is EOFdt or EOFdti. The END Link_Control frame shall be discarded if the End Of Frame delimiter is EOFn, EOFt, EOFni, EOFa or if the end of frame delimiter is missing. Upon the reception of the END Link_Control frame all active Sequences between the communicating N_Ports, using the Class 4 circuit are abnormally terminated.

Exchange and Sequence recovery shall be performed by the communicating N_Ports, at the discretion of the appropriate Upper Level protocol.

N_Ports shall use the END Link_Control frame to remove or deactivate a Class 4 circuit if the N_Port is unable to determine the state of the Class 4 circuit.

The Fabric shall, when required to remove a Class 4 circuit, send END Link_Control commands to both N_Ports in the Class 4 circuit (see 34.2.4.4).

34.7 Login and Service Parameters

The applicability of Class 4 service parameters for Fabric and N_Port Login are shown in tables 98 and 101.

The Fabric shall remove all Class 4 circuits to an N_Port when a Fabric Login is made by the N_Port. Class 4 shall not be used for Fabric Login.

The Class 4 default Fabric Login service parameters shall be:

– Class validity = 0 (invalid).

The state of Class 4 circuits shall not be affected by end-to-end N_Port Login.

The Class 4 default Login service parameters prior to end-to-end N_port Login shall be as shown in table 155.

34.8 Link Services

A Class 4 circuit is setup using Extended Link Service commands associated with Class 4 service.

– QoSR - Quality of Service Request (see 21.18.1)

34.8.1 Basic Link Services

For any Basic Link Service response, if the request is carried on a Class 4 circuit, the VC_ID field shall contain the appropriate CIVC_ID or CRVC_ID value.

34.8.2 Extended Link Services

For any extended link service response, if the request is carried on a Class 4 circuit, the VC_ID field shall contain the appropriate CIVC_ID or CRVC_ID value.

The following Extended Link Service requests shall be supported for Class 4 management (see 21.18):

– Quality of Service Request (QoSR)

– Read Virtual circuit status (RVCS).

34.9 Class 4 procedures

The following procedures specify the behavior for the various phases of a Class 4 circuit.

The flow chart in figure 107 outlines the procedures involved in Class 4 circuit management.

– FLOGI - Each N_Port shall perform Fabric login indicating support for Class 4. (see 34.9.1

– Each N_Port shall perform N_Port login with the QoSF.

– LOGI - Each N_Port shall perform N_Port login with each N_Port for which it intends to establish a Class 4 circuit indicating support for Class 4 (see 34.9.1).

– One of the terminal N_Ports shall start the QoSR link service request with the QoSF. When successful, the Class 4 circuit is placed in the Dormant state (see 34.2.4).

– From the Dormant state, the Class 4 circuit may be activated or reactivated to the Live state by a frame with a circuit activation request between the terminal N_Ports (see 34.2.4.2). The Class 4 circuit may also be removed by the Fabric (see 34.2.4.4).

– While in the Live state, normal interaction between the communicating N_Ports takes place across the Class 4 circuit, moderated by the Fabric.

– When data transfers are complete or are to be suspended, the Class 4 circuit may be moved back to the Dormant state (see 34.2.4.3).

34.9.1 Class 4 login

Before any operations can begin in Class 4, the potential communicating N_Ports shall support Class 4. Their attempt to login with the Fabric provides an indication to the Fabric of their capability to support Class 4 operation and of the Fabric to support Class 4 operation.

If Fabric login ends successfully with Class 4 available to both N_Ports and the Fabric, the N_Ports must then perform N_Port login with each other. Again, the N_Port login procedure indicates the capability of the two N_Ports to support Class 4. If the N_Port login is successful for Class 4, a Class 4 circuit may be setup (see 34.2.4.1).

N_Ports shall Login with the QoSF prior to any QoSR by the N_Port.

34.9.2 Class 4 rules - summary

The following rules apply exclusively to Class 4 circuits. See 34.9.3 for additional rules applicable to the mandatorily supported Intermix with Class 4. To provide a Class 4 circuit, the source and destination N_Ports and the Fabric shall obey the following rules:

a) Except for some Link Service protocols an N_Port shall have performed Fabric Login and N_Port Login with the ports it intends to communicate with, either explicitly or implicitly (see 34.9.1). To login explicitly, the requesting N_Port shall use the Fabric and N_Port Login protocols.

b) The Fabric shall be responsible for Class 4 circuit setup. The Fabric shall retain the setup Class 4 circuit until one of the communicating N_Ports in the Class 4 circuit requests removal of the Class 4 circuit unless one of the legal Fabric exceptions occur. Legal Fabric exceptions includes: path failure and Fabric Login.

c) To activate a Class 4 circuit, the requesting N_Port shall use the Class 4 delimiters as specified in 34.4. An N_Port shall only send an SOFc4 delimited frame on a Class 4 circuit in the Dormant state.

d) To deactivate a Class 4 circuit, the requesting N_Port shall use the Class 4 delimiters as specified in 34.4. The Class 4 circuit may be re-activated at a later time by using the procedure in 34.2.4.2.

e) To remove a Class 4 circuit, the requesting N_Port shall use the Class 4 delimiters as specified in 34.4.

f) A destination N_Port shall acknowledge delivery of each valid Data Frame (see 34.9.4.1).

g) The Sequence Initiator shall increment the SEQ_CNT field on each successive frame transmitted within a Sequence. The Fabric shall guarantee delivery of the frames at the destination N_Port in the order of transmission by the source N_Port.

h) Each N_Port of a Live Class 4 circuit may originate multiple Exchanges and initiate multiple Sequences with one or more N_Ports. Each N_Port shall assign a unique X_ID as appropriate for its role in each Exchange. Thus, the value of the OX_ID and RX_ID is unique for each Exchange with an N_Port pair. a Sequence Initiator shall assign a unique Sequence_Qualifier for each Sequence it initiates which is unique for the communicating N_Port pair.

i) Communicating N_Ports shall be responsible to manage end-to-end flow control without any Fabric involvement. ACK frames shall be used to perform end-to-end flow control (see 34.9.4.1). All Class4 frames, except for the END Link_Control command, shall participate in end-to-end flow control.

j) The Fabric shall be responsible to regulate the rate at which frames are transmitted by an N_Port. It does so by transmitting the VC_RDY Primitive Signal to an N_Port to request that frames be transmitted. Each VC_RDY shall identify the VC for which a frame is being requested. The Fabric shall measure the QoS for bandwidth by measuring all transmission words associated with the VC (SOF delimiters, EOF delimiters, and frame content).

k) The Fabric may reject a request to activate a Class 4 circuit with a valid reason code. Once the Class 4 circuit is alive the QoSF shall only interfere with a VC to the extent that it supplies the VC_RDY Primitive Signals in a timely manner.

l) The Fabric shall respond with an F_BSY Link_Response frame to any received Class 1 connect request frame (i.e., a SOFc1 delimited frame) if one or more Class 4 circuits are setup with the destination N_Port frame is received).

m) The destination N_Port for a request to establish a VC may respond with a P_RJT with an appropriate reason code.

n) An N_Port shall not respond with a P_BSY for any Class 4 frame received.

o) The EE_C4_Credit, determined during the circuit setup process shall be honored.

p) The Fabric shall guarantee that it provides VC_RDY Primitive Signals to the communicating N_Ports at the correct rate and time intervals to meet or exceed the QoS parameters guaranteed for each VC. The latency guarantee relates only to the frame transit within the Fabric.

q) An N_Port shall only transmit Class 4 frames if available VC_Credit permits (see 34.9.4.2).

r) N_Ports shall discard all Class 4 frames received on a Nonexisting Class 4 circuit, or if the Class 4 circuit is in the Pending state.

s) The N_Port shall discard all SOFi4 and SOFn4 delimited frames received on a Class 4 circuit in the Dormant state.

t) An F_Port shall discard all SOFi4 and SOFn4 delimited frames received unless the Class 4 circuit is in the Live state.

u) EE_C4_Credit shall remain unaffected by any discarded Class 4 frames.

v) If VC_Credit is = 0 for a time of more than E_D_TOV the N_Port may use the RVCS protocol to read the state of the Class 4 circuit or the N_Port shall perform the Online to Offline protocol. If the N_Port obtains the Class 4 circuit state from the ACC to the RVCS request and it shows that the Class 4 circuit is Nonexisting, the N_Port shall set the circuit state to Nonexisting.

w) Frames within a Sequence are tracked on a Sequence_Qualifier and SEQ_CNT basis.

x) If an N_Port, not supporting Class 4, receives a Class 4 circuit activation frame, the N_Port may return a P_RJT frame with a reason code of “Class not supported”, or the N_Port may discard the Class 4 frame. If an F_Port, not supporting Class 4, receives a Class 4 circuit activation frame, the F_Port may return a F_RJT response frame with a reason code of “Class not supported”.

y) The Fabric shall use the VC_ID from the frame header for routing frames to a destination N_Port (D_ID). The QoSF maintains a translation from S_ID||VC_ID to D_ID from the original QoSR transaction. Prior to routing a frame, if there is a mismatch on S_ID, VC_ID and D_ID, the frame shall be discarded by the Fabric.

z) An N_Port supporting Class 4 shall not send Class 1 frames nor attempt to establish a Class 1 Dedicated Connection while a Class 4 circuit exists with the N_Port or while a request to setup a Class 4 circuit has been made by the N_Port.

34.9.3 Class 4 intermix

Intermix is always enabled for service Class 4.

– N_Ports may interleave Class 2 or Class 3 frames on the outbound fibre while a Class 4 circuit exists. The same N_Port may receive Class 2 or Class 3 frames on the inbound fibre while a Class 4 circuit exists.

– Transmission of Class 4 frames shall have priority over Class 2 and 3, but a Class 2 or Class 3 frame shall not be aborted to send a Class 4 frame.

34.9.4 Class 4 flow control

34.9.4.1 End-to-end flow control

ACK frames shall be used to perform Class 4 end-to-end flow control. All Class 4 frames shall participate in end-to-end flow control.

NOTE – End-to-end flow control is the only mechanism available for flow control by an N_Ports receiver. A small EE_C4_Credit value helps to decrease peak demands.

34.9.4.2 Buffer-to-buffer flow control

Summary rules for VC credit management:

- N_Ports shall track available credit independently for each VC.
- Maximum available credit (VC_Credit) on a VC shall be limited to a value of less than 256. VC_Credit_Limit < 256)
- An N_Port shall not transmit any Class 4 frames on a given VC unless VC_Credit is available for the VC.
- One credit shall be consumed for every Class 4 frame transmitted on a VC (VC_Credit ⇐ VC_Credit - 1).
- Credit shall be regained when a VC_RDY is received for a given VC (VC_Credit ⇐ VC_Credit + 1 if VC_Credit < VC_Credit_Limit)

NOTE – Limiting available credit allows the Fabric to generate more VC_RDYs than the number of frames transmitted. Allowing unused bandwidth to be used by other VCs or for intermix frames.

The state of a VC shall be the same as the Class 4 circuit state for that VC. The maximum available credit (VC_Credit) shall be limited as follow:

- For the CTR to the value of the Live VC credit limit field in the QoSR, if the VC is alive.
- For the CTI to the value of the Live VC credit limit field in the accept frame to the QoSR, if the VC is alive.
- Maximum available credit shall be 0 for Nonexisting VC or VCs in the Pending state. (VC_Credit_Limit = 0)

NOTE – The CTI and the CTR start with an available VC_Credit of 0 when a new VC is setup.

- Maximum available credit shall be 1 for VCs in the Dormant state. (VC_Credit_Limit = 1)
- Maximum available credit shall be set to the Live_VC_Credit_Limit for VCs in the Live state.
- Available VC_Credit shall be 0 after Link Reset. (VC_Credit ⇐ 0)

If the N_Port has a frame ready to transmit and EE_C4_Credit available when a VC_RDY is received for a VC, the N_Port transmits the frame at the next opportunity to transmit an outbound frame. The N_Port may be in the middle of transmitting a frame when a VC_RDY is received.

The N_Port shall not accumulate a VC_RDY buffer-to-buffer credit that exceeds the allotted limit. The N_Port shall not use a count in excess of the preset limit to send several frames.

Receipt of a VC_RDY is a guarantee by the Fabric that bandwidth is available to transmit a frame for the identified VC.

The rate at which the Fabric provides VC_RDYs to an N_Port can be used to provide congestion control for the Fabric by keeping unwanted frames in the N_Ports until the QoS parameters can be met.

34.10 Quality of Service

Class 4 through the QoSF, guarantees that the minimum bandwidth is available to the communicating N_Ports in a Class 4 circuit. Some applications may cause the communicating N_Ports not to use all of that guaranteed bandwidth, but the Fabric shall offer the guaranteed bandwidth to each N_Port for each Live Class 4 circuit.

Therefore, the measure of effectiveness of Class 4 in meeting its goal of guaranteed bandwidth cannot be measured for a Fabric without consideration for the participating N_Ports. The measure of effectiveness can only be obtained indirectly by the VC_RDY Primitive Signals transmitted to the participating N_Ports. The bandwidth utilization for a VC is also a function of the available EE_C4_Credit and the ability of the participating ports to sustain the negotiated bandwidth.

The Fabric shall guarantee a worst case end-to-end delay for each frame once it enters the Fabric.

To help establish a frame rate associated with a calculation of the negotiated bandwidth, the Quality of Service Extended Link Service requests permit the N_Ports to negotiate a maximum frame Data Field size that is equal to or smaller than the current allowable maximum between the communicating N_Ports specifically for Class 4. The maximum value is determined during Fabric and N_Port Login.

All QoS parameters are negotiated through the QoSF for each VC. The QoS parameters are independently negotiated for each VC. This permits an asymmetric allocation of resources.

- Bandwidth range
- Worst case end-to-end delay for a frame for fractional bandwidth mode
- VC Data Field size.

For the first parameter there are two levels of negotiation. The desirable quality of service (Desirable QoS) and the acceptable quality of service (Acceptable QoS).

The Acceptable QoS is guaranteed if the Class 4 circuit is setup. The desirable QoS may be attempted, but is not guaranteed. When applied to the list above, the negotiable parameters become:

- Desirable Bandwidth
- Acceptable Bandwidth
- Acceptable Worst case end-to-end delay for a frame
- VC Data Field size.

The VC Data Field size is not a variable quantity since it represents the maximum Data field size that an FC-2 can generate for each frame for Class 4 and is not controllable by the Fabric.

The negotiation for QoS is accomplished using an Extended Link Service request, QoS Request (QoSR) (see 21.18.1).

Each request is handled in two stages: CTI to QoSF and QoSF to CTR. This process is illustrated in the specification of the QOSR link service request.

- If the QoSF cannot perform the function for the Class 4 circuit (e.g., too much bandwidth requested versus that available at the time of the request) the QoSF shall send a LS_RJT response to the CTI with an appropriate reason code.
- If the CTR cannot perform the function for the Class 4 circuit (e.g., too much bandwidth requested versus that available at the time of the request) the CTR shall send an LS_RJT to the CTI to inform it of the rejection and the reason for it.

35 Camp-On

35.1 Introduction

Camp-On introduces a level of fairness into the connection establish mechanism that is enforced by the Fabric. It ensures the Connection Initiator (CI) eventual access to the destination N_Port. The Camp-On protocol requires active participation by the CI and the Fabric while the Connection Recepient (CR) has a passive role in the Camp-On connect request protocol.

Camp-On provides an N_Port the means to alter the behavior of the Fabric during the connection request process. Camp-On allows the source N_Port to get on a queue at the destination F_Port when the destination N_Port is engaged in a connection and thus be assured of eventual access to the destination N_Port. Camp-On is invoked on a connection by connection basis.

The Connection Initiator (CI) can direct the Fabric to immediately (within E_D_TOV) ‘busy’ the connect request if the destination N_Port is engaged in a connection or to request that the Fabric ‘hold’ the connection request for an extended time period.

NOTE - No new delimiters are required.

35.2 FC-2 Mechanisms

Enhanced FC-2 mechanisms used for Camp-On are:

- Camp-On Request (COR): CS_CTL bit (see 18.2)
- Notify sequence protocol (see 20.3.4.2)
- End Link Control Command

35.3 Applicability

Camp-On is limited to Class 1 service. Camp-On is an optional function that may be supported by the CI and the Fabric. The CR is not aware of the function. It is related to Stacked Connection Request which is also limited to Class 1 service.

35.4 Communications Model

The Connection Initiator (CI) N_Port requesting a connection will set or reset the COR bit in the CS_CTL field of the connection request to invoke or not to invoke the Camp-On function in the Fabric. If the COR bit is reset (=0) and if the destination N_Port is engaged in a connection, the Fabric will immediately (within E_D_TOV) busy off the connection request. When the COR bit is set (=1) and if the destination N-Port is currently engaged in a connection, the Fabric will place the connection request on a queue for that destination F_Port. If that destination N_Port supports Intermix, the Fabric will initiate a Class_2 Notify (see 20.3.4.2) sequence to that N_Port. If the destination N_Port does not support Intermix, the Fabric shall not transmit the Notify sequence. If the destination N_Port is not engaged in a connection when the connect request is received by the Fabric, the Fabric will immediately initiate the connection process (normal operation).

If the Fabric is unable to service the connect request due to any Fabric conditions such as queue being full, the Fabric shall transmit an F_BSY to the CI and discard the connect request.

If the CI does not receive F_BSY nor get the connection established and if the CI chooses not to wait any longer, the CI may send an End Link Control command with EOFdt and the Fabric shall remove the connect request off the queue and discard it. If the Fabric has already routed the connect request to the CR, then the Fabric shall transmit an F_RJT.

35.5 Requirements

35.5.1 Connection Initiator

The CI shall support:

- Class 1
- COR CS_CTL bit
- CI rules stated below.

35.5.2 Connection Recipient

The CR may optionally support:

- Notify protocol.

35.5.3 Fabric

The Fabric shall support

- Class 1
- Intermix

– Notify protocol.

– COR CS_CTL bit

35.6 Login

N_Ports and Fabric shall indicate their ability to support Camp-On during Fabric Login.

35.6.1 F_Port Class Service Parameters

The F_Port Class Service Parameters (FC-PH 23.7.4) shall be used to indicate support for the Camp-On facility (see 23.7.4.2).

35.7 Camp-On Invoked

The CI shall invoke the Camp-On function on a connection by connection basis. To invoke Camp-On, the CI shall transmit a connect request frame containing the SOFc1 delimiter and setting the CS_CTL COR=1. If COR=0 on a connect request frame, the Camp-On function is not invoked.

35.8 Rules

35.8.1 Connection Initiator

– To invoke Camp-On, the CI shall transmit a connect request frame containing the SOFc1 delimiter and setting the CS_CTL COR=1.

– When the CI receives an F_BSY in response to a connect request frame with COR set to a one (=1), it shall know that its connect request has been discarded and the requested connection will not be established. It must retransmit the connect request at a later time.

– If neither the connection has been established, nor an F_BSY has been received and if the CI chooses not to wait any longer, the CI may transmit an End Link Control command with EOFdt. Through the End Link Control command, the CI is indicating that the connect request shall be discarded by the Fabric.

Table 156 – Camp-On vs Stacked Connect Request

	Stacked Connect Request (SCR)	**Camp-On**
Support Indicated:	by Fabric during Login	by N_Port & Fabric during Login
Invoked by	CI through SCR CS_CTL on a connection by connection basis	CI through COR CS_CTL on a connection by connection basis
Response Received to Connect Request		
with S_ID=A and	D_ID=B, C, D,...	D_ID=B
ACK with S_ID=B	- Connection established with one of the desired destinations	- same as SCR
SOF(Cx) with S_ID=X	- Connect est. with third party - Connect Request maintained within Fabric	- Connection est.with third party - Connect request failed - Retransmit Connect Request
F_BSY	Retransmit Connect Request	same as SCR
F_RJT	Retransmit Connect Request	same as SCR
P_BSY	Retransmit Connect Request	same as SCR
P_RJT	Retransmit Connect Request	same as SCR
CR_TOV use	- CR_TOV + E_D_TOV or 2 (CR_TOV) + E_D_TOV - Connection could not be established within time out period. - Retransmit Connect Request	Not applicable

35.8.2 Connection Recipient

– The CR shall transmit all ACK, P_BSY, and P_RJT frames in the appropriate class of service according to the state of the Simplex CS_CTL bit in the connect request frame:

– if Dedicated Duplex (Simplex=0), the CR shall use Class 1 SOF delimiters.

– if Dedicated Simplex (Simplex=1), the CR shall use Class 2 SOF delimiters.

– If the CR receives the NTY Link Control command (see 20.3.4.2), the CR shall end the existing connection by initiating the End-Connection protocol as defined in sections 18.5 and 28.4.3 of FC-PH.

35.8.3 Fabric

– The Fabric shall transmit all ACK, F_BSY, and F_RJT frames in the appropriate class of service according to the state of the Simplex CS_CTL bit in the connect request frame:

– If Dedicated Duplex (Simplex=0), the Fabric shall use Class 1 SOF delimiters.

– If Dedicated Simplex (Simplex=1), the Fabric shall use Class 2 SOF delimiters.

– The Fabric shall transmit the NTY Link_ Command frame when Camp-On connect request is queued waiting for the destination N_Port to become available if the destination N_Port supports intermix.

– If the Fabric is unable to service the connect request due to Fabric conditions such as queue full, the Fabric shall discard the connect request frame, transmit the F_BSY as the response to the connect request, and take no further action to make the connection.The F_BSY shall use the appropriate SOF delimiter and an EOFt delimiter.

– If the Fabric receives the End Link Control command and has not routed the connect request to the CR, the Fabric shall remove the connect request off the queue and discard it.

– If the Fabric receives the End Link Control command but has already routed the connect request to the CR, the Fabric shall issue a F_RJT and establish the connection when it becomes possible.

35.8.4 Multiple requests

– If both SCR and Camp-On requests are indicated for the same connection in SOFc1, then the Fabric shall reject the request by issuing a F_RJT with a reason code = protocol error.

36 Stacked Connect Request

36.1 Introduction

The Stacked Connect Request (SCR) function as described in FC-PH Clauses 23 and 28 are modified by this Clause.

The ability to support Stacked Connect Request is indicated by a Fabric during the Fabric Login procedure and invoked by an N_Port through the CS_CTL SCR bit of the connect request frame. It is invoked on a connection by connection basis. The CR_TOV timer defined in Clause 29 is used by the Connection Initiator (CI), the Connection Recipient (CR), and the Fabric to time the connection establishment procedure.

NOTE - No new delimiters are required.

36.2 FC-2 Mechanisms

The FC-2 mechanisms used for this function are:

– SCR: Stacked Connect Request

– Class Specific Control field (CS_CTL)

– CR_TOV: Connection request time out value.

36.3 Applicability

The Stacked Connect Request is an optional function. If this function is invoked, the use of the CR_TOV is mandatory.

SCR is applicable to Class 1 Dedicated Duplex and Dedicated Simplex connections.

36.4 Communications Model

The ability of the Fabric to support Stacked Connect Request is indicated during the Fabric Login procedure through the service parameters supplied by the Fabric. The SCR function is invoked by the CI through the SCR CS_CTL bit in the Frame Header of the connect request frame. Thus the SCR function is invoked on a connection by connection basis under control of the CI. This is a departure from the SCR definition in FC-PH which defined SCR as a port function invoked for all connect request during Login.

36.5 Requirements

36.5.1 Connection Initiator

The CI shall support:

– Class 1

– SCR CS_CTL bit

– CR_TOV timer.

– SCR in Login

36.5.2 Connection Recipient

The CR shall support:

– Class 1

– CR_TOV timer.

36.5.3 Fabric

The Fabric shall support

– Class 1

– SCR CS_CTL bit

– CR_TOV timer.

– SCR in Login

36.6 Login

The negotiation process described in FC-PH 23.6 is replaced (see 23.7.4.2 and Table 102).

The ability to support the SCR function by the Fabric is indicated during Fabric Login in ithe F_Port Class Service parameters (see 23.7.4.2), as stated in FC-PH 23.7. The N_Port service parameters interchanged during Fabric Login (see 23.6.7.2) for the SCR function are not used.

36.7 SCR Invoked

The CI shall invoke the Stacked Connect Request function on a connection by connection basis. To invoke SCR, the CI shall transmit a connect request frame containing the SOFc1 delimiter and setting the CS_CTL SCR=1. If SCR=0 on a connect request frame, the SCR function is not invoked.

36.8 Timer

Stacked Connect Request utilizes both the CR_TOV timer and the E_D_TOV timer.

36.8.1 CR_TOV

The CR_TOV timer shall be used by the Fabric to define the maximum time period that the Fabric shall hold a connect request frame when SCR is invoked and the destination N_Port is engaged in a prior connect. The Fabric shall attempt to establish the requested connection within the CR_TOV time out period. If the CR_TOV timer expires before the Fabric transmits the connect request to the destination N_Port, the Fabric shall discard the connect request and transmit a F_BSY to the source N_Port.

36.8.2 E_D_TOV

The E_D_TOV timer shall be used by the Fabric to define the maximum time period that the Fabric may hold a connect request when SCR is not invoked.

36.9 Rules

36.9.1 Connection Initiator

– To invoke SCR, the CI shall transmit a connect request frame containing the SOFc1 delimiter and setting the CS_CTL SCR=1.

– If the Fabric indicated support for SCR in Fabric Login (see 23.7.4.2 and table 102), the CI shall wait a time out period as specified in 29.2.11.1.

36.9.2 Connection Recipient

– The CR shall wait a time out period as specified in 29.2.11.2.

36.9.3 Fabric

– If a Fabric received a connect request frame with the SCR CS_CTL bit 30 = 1, it shall invoke the rules for the SCR mode indicated in its service parameters during Fabric Login (see 23.7.4.2 and table 102).

– The Fabric shall manage CR_TOV timer as specified in 29.2.11.3.

36.9.4 Multiple requests

– If both SCR and Camp-On requests are indicated for the same connection in SOFc1, then the Fabric shall reject the request by issuing a F_RJT with a reason code = protocol error.

37 Buffered Class 1 Service

Buffered Class 1 extends Class 1 Service to allow N_Ports of differing data rates to establish and make use of Class 1 Service. However, only maximal bandwidth rather than full bandwidth is guaranteed between the N_Ports. The data rate translator facility, if present, shall be located in the Fabric. The Fabric shall not preclude the use of Buffered Class 1 Service between equal data rate N_Ports, nor shall the availability of Buffered Class 1 Service preclude normal unbuffered Class 1 usage. Buffered Class 1 may be used in combination with any of the other Class 1 features, i.e. both types of Stacked Connect Request, Camp-On, Dedicated Simplex and Unidirectional or Bidirectional Duplex type connections.

When Buffered Class 1 Service is enabled, an N_Port can freely elect to use or not use Buffered Class 1 each time a Class 1 Connection is established. Buffered and unbuffered Class 1 connect requests can be mixed without any restrictions even if Stacked Connect Requests or Camp-On is enabled.

37.1 Fabric Login

The support for Buffered Class 1 Service is negotiated during Fabric Login (see 23.3). If an N_Port requests Buffered Class 1 Service and the attached F_Port supports it, only then is Buffered Class 1 Service enabled and available for use by the N_Port and Fabric.

Buffered Class 1 uses BB_Credit rules between the N_Port and the F_Port. The BB_Credit rules applies to all Class 1 frames transmitted by an N_Port, on an established Buffered Class 1 connection, but do not apply to any SOFi1 and SOFn1 delimited frames transmitted by the F_Port. Neither the Fabric nor the F_Port shall restrict the frame size of Class 1 SOFi1 and SOFn1 delimited frames below the 2148 byte maximum frame size limit specified in FC–PH.

37.2 Applicability

Buffered Class 1 is an optional service that may be provided by the Fabric and N_Ports.

37.2.1 Topology

Buffered Class 1 is applicable to the following topologies:

- Fabric;
- Fabric attached Arbitrated Loop.

Both the point-to-point and the stand-alone Arbitrated Loop topology are restricted to a common data rate and therefore do not require Buffered Class 1 service.

37.3 Fabric guaranteed bandwidth

Buffered Class 1 Service provides a maximal bandwidth guarantee across the Dedicated Simplex or Duplex connection. This guarantee is ensured by the following:

- The internal Fabric path, used for an established Buffered Class 1 Dedicated connection shall not be used to carry any Class 2, 3 or 4 frames while the Buffered Class 1 Dedication connection exists.
- As described in FC-PH, if intermix is enabled, shall Class 1 frames have priority over Class 2 and 3 Intermix frames.

37.4 Delimiters

Buffered Class 1 does not affect delimiter usage of the Class 1 service.

37.5 CS_CTL bit

CS_CTL bit 28 is used for Buffered Class 1 Request (BCR) (see 37.6).

If bit 28 in the CS_CTL field is = 1 then Buffered Class 1 is requested. If bit 28 in the CS_CTL field is = 0 then unbuffered, full bandwidth, Class 1 is requested. This bit is meaningful on all Class 1 frames.

37.6 Rules

The following rules apply specifically to Buffered Class 1. However, all the FC-PH and FC-PH-2 defined rules for Class 1 also apply. In case of conflict, the rules defined in this clause shall take precedence over the rules defined in FC-PH or elsewhere in FC-PH-2. Examples of the Class 1 frame and R_RDY flow for both Buffered and unbuffered Class 1 are shown in figure 108.

37.6.1 Connection_Initiator N_Port

- The N_Port shall set the Buffered Class 1 Request (BCR) bit in the CS_CTL field = 1 on

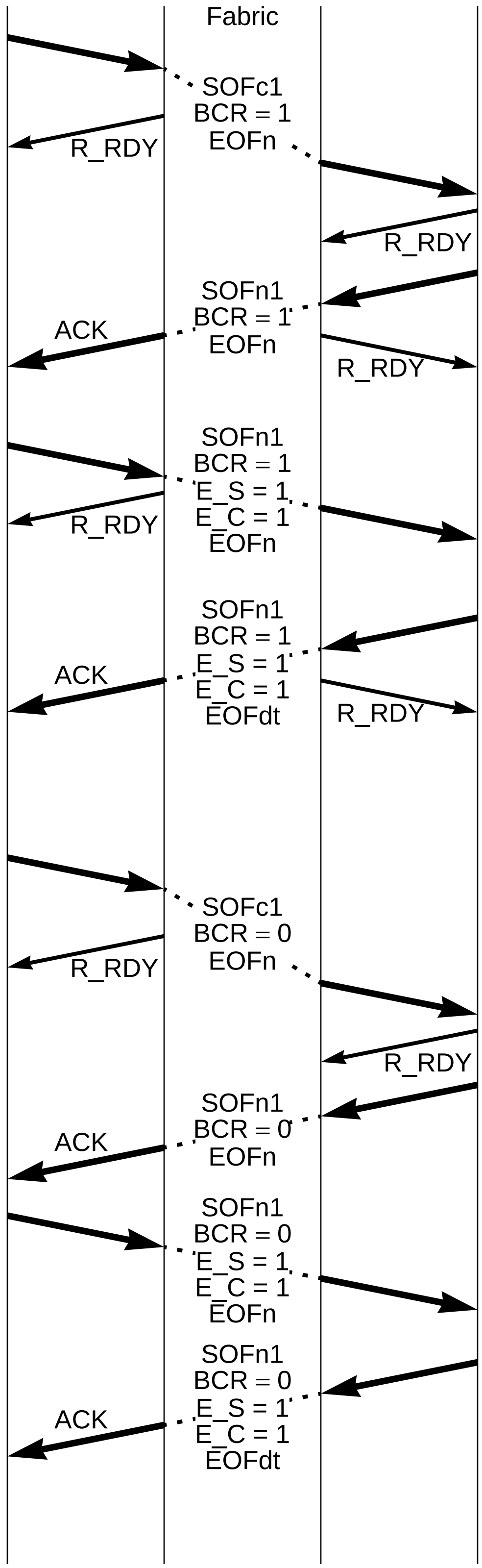


Figure 108 – Buffered and unbuffered Class 1 frame and R_RDY flow

the Class 1 connect frame transmitted to establish a Buffered Class 1 connection.

– The N_Port shall set the BCR bit in the CS_CTL field = 1 on all Class 1 frames transmitted on an established Buffered Class 1 connection.

– The N_Port shall only transmit Class 1 frames on an established Buffered Class 1 connection if permitted by BB_Credit rules (see 37.6.4).

– The N_Port shall obey all other rules defined by FC-PH or FC-PH-2 for the established Class 1 connection type, i.e. the Unidirectional Duplex, Bidirectional Duplex or Dedicated Simplex rules.

37.6.2 Connection_Recipient N_Port

– The N_Port shall set the BCR bit in the CS_CTL field = 1 on all Class 1 frames transmitted on an established Buffered Class 1 connection.

– The N_Port shall only transmit Class 1 frames on an established Buffered Class 1 connection if permitted by BB_Credit rules.

– The N_Port shall obey all other rules defined by FC-PH or FC-PH-2 for the established Class 1 connection type, i.e. the Unidirectional Duplex, Bidirectional Duplex or Dedicated Simplex rules.

37.6.3 F_Port

– The Fabric shall F_RJT any Buffered Class 1 Connect requests, SOFc1 delimited frames, destined for N_Ports for which Buffered Class 1 is not enabled. The reason code shall be 'Fabric path not available'. This ensures compatibility with FC-PH compliant N_Ports.

– The Fabric shall discard any SOFi1 or SOFn1 delimited frames on an established Class 1 connection, if the BCR bit in the CS_CTL field is set = 1 and Buffered Class 1 is enabled but unbuffered Class 1 was indicated by the Connect frame.

– The Fabric may discard any SOFi1 or SOFn1 delimited frames on an established Buffered Class 1 connection, if the BCR bit in the CS_CTL field is set = 0.

– The Fabric may discard any SOFi1 or SOFn1 delimited frames on an established Buffered Class 1 connection, if BB_Credit rules are violated (see 37.6.4).

– The Fabric may F_RJT any Class 1 connect frames received from an N_Port for which Buffered Class 1 was not enabled. If the Fabric rejects the frame the reason code shall be ‘Invalid CS_CTL field’.

– The Fabric response is unpredictable if an SOFi1 or SOFn1 delimited frames is received in which the BCR bit in the CS_CTL field is set = 1 if Buffered Class 1 is not enabled.

– The F_Port shall return an R_RDY for every SOFi1 or SOFn1 delimited frame received on an established Buffered Class 1 connection,. The R_RDY shall be transmitted as receive buffers becomes available.

– SOFi1 and SOFn1 delimited frames received on an established Buffered Class 1 connection shall be forwarded by the Fabric to the destination N_Port unless discarded for one of the above reasons. No address validation or routing shall be performed on these frames by the Fabric. However, frame validation as specified by FC-PH clause 17.6.2 shall be performed.

37.6.4 BB_Credit rules for Buffered Class 1

BB_Credit rules are largely unaffected by Buffered Class 1, except that an N_Port follows BB_Credit rules when it transmits SOFi1 and SOFn1 delimited frames on an established Buffered Class 1 connection. The BB_Credit rules for Buffered Class 1 are:

– N_Ports shall only transmit SOFi1 or SOFn1 delimited frames on established Buffered Class 1 connections if BB_Credit allows.

– Transmission of an SOFi1 or SOFn1 delimited frames on established Buffered Class 1 connections shall consume 1 BB_Credit.

– Reception of an R_RDY shall reclaim 1 BB_Credit.

– Unless the alternative credit model is in effect, as specified in FC-AL, the BB_Credit shall be limited to value obtained during Fabric Login.

37.6.5 Buffered Class 1 frame size

The rules determining Class 1 frame size shall remain unaffected by Buffered Class 1. The frame size rules for Buffered Class 1 are summarized below:

– SOFc1 delimited frames are limited in size to the lesser of the Fabric or the recipient N_Ports Receive Data_Field size.

– SOFi1 or SOFn1 delimited frames are limited in size only by the recipient N_Ports Receive Data_Field size.

– The Fabric shall truncate SOFi1 or SOFn1 delimited frames longer than the maximum allowed, 2112 Byte Receive Data_Field size. Truncated frames shall be terminated by an EOFa end of frame delimiter.

37.6.6 Intermix

Intermix rules and behavior shall remain unaffected by Buffered Class 1. FC-PH defined Class 2 and 3 frames may freely be intermixed if Intermix is enabled.

38 Data compression

This clause describes and formally defines the Adaptive Lossless Data Compression (ALDC) method and compressed data stream format.

38.1 Introduction

Lossless data compression can be provided by implementing the ALDC LZ-1(Adaptive Lossless Data Compression Lempel Ziv-1) algorithm (see ALDC description) as an optional FC-2 function. The option can be invoked by the Initiator on a per Information Category basis within a Sequence, immediately preceding segmentation to frame Payloads. A separate single continuous history (see ALDC description) should be used for each Information Category within a Sequence. Decompression should be performed at the recipient immediately following Information Category reassembly.

38.2 N_Port Login

The Sequence Initiator's capability to perform data compression and the Sequence Recipient's capability to perform decompression are determined during N_Port Login.

Only if both Sequence Initiator and the Sequence Recipient support the capability, they can perform the function.

38.2.1 Initiator Capability

To the Initiator Control Flags (D) specified in FC-PH 23.6.8.3, additional flags are specified in FC-PH-2 (see 23.6.8.3).

The data compression capability is indicated by the Sequence Initiator during N_Port Login in word 0, bit 8 of N_Port Class service Parameters under Initiator Control.

Word 0, bits 7-6 are defined to indicate the History buffer size supported by the Initiator (see 23.6.8.3).

The Initiator shall use the largest buffer size common with the Recipient, indicated in the Recipient control flags specified in 23.6.8.4.

38.2.2 Recipient Capability

To the Recipient Control Flags (D) specified in FC-PH 23.6.8.4 additional flags are specified in FC-PH-2 (see 23.6.8.4).

The data decompression capability is indicated by the Sequence Recipient during N_Port Login in word 1, bit 23 of N_Port Class service Parameters under Recipient Control.

Word 1, bits 22 - 21 are defined to indicate the History buffer size supported by the Recipient (see 23.6.8.4).

The Recipient shall use the largest buffer size common with the Initiator, indicated in the Initiator control flags specified in 38.2.1.

38.2.3 F_CTL

Data compression status of an Information Category shall be indicated by the Sequence Initiator to the Sequence Recipient by means of F_CTL bit 11 (see 18.5).

38.3 Applicability

Data compression is applicable to all Classes. Data compression status (F_CTL bit 11) is meaningful in all Data frames of an Information Category.

NOTE - F_CTL bit is defined to be meaningful in all Data frames to ensure that the Sequence Recipient recognizes compressed data with RO usage and out-of-order frame delivery.

38.4 Decompression

The Sequence Recipient shall decompress data on a per Information Category basis, using Information Category bits of R_CTL and the SEQ_ID of the Sequence to which the Information Category belongs.

SInce the Data compression is performed on a per Information Category basis (not on frame by basis), the RO shall not be used with Data compression. If the Sequence Recipient uses RO and transmits compressed data, the Sequence Recipient shall ignore the RO and decompress the data for the entire Information Category.

38.5 Algorithm overview

The ALDC algorithm is one variant of the LZ_1 (Lempel-Ziv 1) class of data compression algorithms, first proposed by Abraham Lempel and Jacob Ziv in 1977 [1].

LZ_1 algorithms achieve compression by building and maintaining a datastructure, called a HISTORY_BUFFER. An LZ_1 encode process and an LZ_1 decode process both initialize this structure to the same known state, and update it in an identical fashion. The encoder does this using the input data it receives for compression, while the decoder generates an identical data stream as its output, which it also uses for the update process. Consequently, these two histories can remain identical, so it is never usually necessary to include history content information with the compressed data stream sent from an encoder to a decoder. Usually, for general purpose use, LZ_1 encoders or decoders both reset their history structures to a cleared or empty state. Incoming data is copied into the history, and is initially encoded explicitly. However, as the history fills, it becomes increasingly possible for the encoder to represent incoming data by encoding a reference to a string already present in this history. This is the principal mechanism by which LZ_1 algorithms are able to achieve compression.

38.6 Encoder and decoder operation

An encoder processes incoming data 1 byte at a time. Each byte of data processed is always copied in the order received to the history, the oldest data being displaced if the history is already full. Thus a sequential copy of the most recently processed data is always available.

The compression process consists of examining the incoming data stream to identify any strings of data bytes which already exist in the encoder history. If an identical such history is available to a decoder, this matching string can be encoded and output as a 2 element COPY_POINTER, containing a byte count and history location. It is then possible for a decoder to reproduce this string exactly, by copying it from the given location in its own history. If the COPY_POINTER can be encoded in fewer bits of information than required for the data string it specifies, compression is achieved.

If an incoming byte of data does NOT form part of a matching string, a LITERAL, containing this embedded value, is encoded and then output to explicitly represent this byte.

A decoder performs the inverse operation by first parsing a compressed data stream into LITERALS and COPY_POINTERS for processing.

A LITERAL is processed simply by determining the data byte value which is embedded in it. This value is then output as the next data byte and is also copied into the decoder history. The decoder history will then once more be identical to that existing within the encoder.

A COPY_POINTER is processed by first decoding the specified byte count and history location elements of the pointer. Then this string of data byte values is read, one byte at a time, from the decoder history. The data byte values are each output as decoded data, and also copied into the history, before the next value is accessed. Once the entire string has been processed in this manner, the decoder history is identical to that in the encoder once more.

38.7 Derivation of ALDC encoding structure

The many LZ_1 variants differ mainly in the size of history structure, and in the encoding method used to represent literals and pointers.

The original scheme published by Lempel and Ziv [1] used a series of triples as its output. These can be regarded as an ordered triples of strictly alternating history references and characters, the history reference requiring two values of the triple and the character one.

In the literature, this scheme is commonly referred to as LZ77, and in terms of the nomenclature introduced above, an alternating order of COPY_POINTERS and LITERALS is always generated as its output.

In Bell's 1986 paper [3] he implements a suggestion made earlier by Storer and Szymanski in 1982 [2], in which it is proposed that the Lempel-Ziv algorithms use a free mixture of these history pointers and literal characters, literals only being used when a history pointer takes up more space than the characters it codes.

Bell's implementation of this scheme is called LZSS, and adds an extra bit to each pointer or character to distinguish between them. The ALDC encoding uses this same principle exactly, thus each LITERAL is always 9 bits in length, consisting of a single 0 bit, followed by the 8 data bits of the embedded character value.

The 1987 paper by Brent [4] describes the use of hashing to implement the history string match process rapidly in software, and also uses variable length codes to efficiently encode both the literal values and the ordered value pairs constituting history references.

Brent's implementation of this scheme is called SLH, and it uses a two pass technique, in which the literal values and the history references are first determined. An optimal Huffman code is then generated, based on the statistical frequencies of these values, and used to encode the output in a second pass. Brent also describes the use of a single pass method in which an adaptive Huffman or Arithmetic coding technique can be used to encode the literals and history references as they occur.

ALDC implementations use an exhaustive search, that is, every possible candidate is considered for string matching operations. Empirically, a fairly uniform distribution of COPY_POINTER location values results if this is done, while the length values tend to approximate closely to a logarithmic distribution. This result is observed over a wide range of types of data commonly stored or manipulated by computer systems.

Consequently, ALDC uses a non-adaptive coding method for these values, the history location being encoded on a fixed length binary field, and the length values on a quasi-logarithmic code, these being the optimal theoretical solutions for encoding these kinds of distributions.

The length code used in ALDC varies in size from 2 to 12 bits, and can represent a range of 286 values in total. The last 16 consecutive code values in the range, all 12-bit numbers, are reserved for control use, and at this time only the uppermost code is defined. This is used as a marker to indicate the end of a compressed data stream.

There are thus 270 length codes available, which allow encoding of the matching string lengths of from 2 through 271 bytes. Matching single byte strings are not encoded as COPY_POINTERS in ALDC, since no compression would result [3].

ALDC history structures shall be reset to an empty, or cleared state at the start of an operation. Data is always stored in ascending location order to these history structures, starting with a location address of zero. This location, or displacement value, shall reset to zero each time the entire history is filled, and the update location is NOT used for either initiating or continuing string matching operations.

38.8 ALDC History_Buffer Sizes

ALDC is defined for three different sizes of HISTORY_BUFFER, expressed in multiples of 512 byte units represented by a suffix. Thus ALDC with a 512 byte history structure is referred to as ALDC_1, if the size for the history structure is 1024 bytes, it is denoted by ALDC_2, while if a 2048 byte history structure is provided, it is denoted by ALDC_4.

The history location field in the COPY_POINTER is a binary field which simply contains the displacement in the history structure of the start of the referenced string, and its size is therefore 9 bits for ALDC_1, 10 for ALDC_2 and 11 bits for ALDC_4 implementations.

NOTE - Compression in general increases with larger history structures, but a typical increase in compression ratio is about 3% for doubling history structure size from 512 to 1024 bytes, and from 1024 to 2048 bytes. In the case of hardware implementations, however, chip area and power are roughly doubled each time. The 512 byte history structure size is thus recommended as providing the best trade-off.

38.9 Rules

The following rules apply to Data compression and decompression:

1. The Sequence Initiator is responsible for data compression and may choose its own criteria as to which Information Categories to compress.
2. The Sequence Initiator shall indicate its ability to perform data compression during N_Port Login (see 38.2.1).
3. The Sequence Initiator may transmit compressed data, only if the Sequence Recipient has indicated during N_Port Login, its ability to perform decompression.
4. If the Sequence Initiator transmits compressed data to a Sequence Recipient which does not have the decompression capability, the Sequence Recipient shall issue a BA_RJT.
5. The Sequence Initiator shall set F_CTL bit 11 =1 in all Data frames of the Information Category whose data is compressed (see 18.5).
6. The Sequence Initiator shall use a single continuous history for each Information Category within a Sequence.
7. The history buffer shall be reset to an empty state and a new history started for each Information Category to be compressed.
8. The Sequence Recipient is responsible for decompression and it shall indicate its ability to perform decompression during N_Port Login (see 38.2.2).
9. If the Sequence Recipient receives frames belonging to an Information Category out-of-order, it shall use the SEQ_CNT to reorder the frames before decompressing the entire Category.
10. If the Sequence Recipient receives a frame with a valid RO with F_CTL bit 11 =1, it shall ignore the RO value and perform the decompression for the entire Information Category based on the SEQ_CNT.

38.10 Formal ALDC Format definition

The compressed data stream encoding format is described using BNF-like language, the symbols being defined as follows:

Table 157 – Symbols

<table>
<tr><th>Symbol</th><th>Definition</th></tr>
<tr><td>:=</td><td>the non-terminal on the left side of the ":=" can be replaced by the expression on the right side</td></tr>
<tr><td><name></td><td>non-terminal expression</td></tr>
<tr><td>[ ]</td><td>the expression inside the "[ ]" may occur 0 or more times</td></tr>
<tr><td>|</td><td>the logical disjunction "or"</td></tr>
<tr><td>( )</td><td>the text enclosed within the ( ) is a comment only, added for clarity</td></tr>
<tr><td>0,1</td><td>the terminal binary digits 0 or 1</td></tr>
</table>

38.10.1 ALDC_1 Algorithm - Formal definition (512 byte history)

The compressed data for 512 byte history is defined by the following expression:

<Compressed_Data> := [0 <LITERAL> | 1 <COPY_PTR>] 1<CONTROL>

where <LITERAL> := <b><b><b><b><b><b><b><b> (8-bit byte data)

<b> := 0 | 1

<COPY_PTR> := <length_code><displacement>

<length_code> := 2, 4, 6, 8 or 12 bits

<displacement> := <b><b><b><b><b><b><b><b><b> (9-bit)

<CONTROL> := (ctl_code)

The length_code and ctl_code are specified in tables 158 and 159.

<table>
<tr><th colspan="3">Table 158 – Length_code</th></tr>
<tr><td>(length_code)</td><td>(field value)</td><td>(COPY_PTR length)</td></tr>
<tr><td>00</td><td>(0)</td><td>(2 bytes) |</td></tr>
<tr><td>01</td><td>(1)</td><td>(3 bytes) |</td></tr>
<tr><td>10 00</td><td>(2)</td><td></td></tr>
<tr><td>: : :</td><td>: : :</td><td>: : :</td></tr>
<tr><td>10 11</td><td>(5)</td><td>(7 bytes) |</td></tr>
<tr><td>110 000</td><td>(6)</td><td>(8 bytes) |</td></tr>
<tr><td>: : :</td><td>: : :</td><td>: : :</td></tr>
<tr><td>110 111</td><td>(13)</td><td>(15 bytes) |</td></tr>
<tr><td>1110 0000</td><td>(14)</td><td>(16 bytes) |</td></tr>
<tr><td>: : :</td><td>: : :</td><td>: : :</td></tr>
<tr><td>1110 1111</td><td>(29)</td><td>(31 bytes) |</td></tr>
<tr><td>1111 0000 0000</td><td>(30)</td><td>(32 bytes) |</td></tr>
<tr><td>: : :</td><td>: : :</td><td>: : :</td></tr>
<tr><td>1111 1110 1110</td><td>(268)</td><td>(270 bytes) |</td></tr>
<tr><td>1111 1110 1111</td><td>(269)</td><td>(271 bytes)</td></tr>
</table>

<table>
<tr><th colspan="3">Table 159 – Ctl_code</th></tr>
<tr><th>(ctl_code)</th><th>(field value)</th><th>(control specified)</th></tr>
<tr><td>1111 1111 0000</td><td>(270)</td><td rowspan="3">the 12 bit field values of 270 to 284 are reserved</td></tr>
<tr><td>: : :</td><td>: : :</td></tr>
<tr><td>1111 1111 1110</td><td>(284)</td></tr>
<tr><td>1111 1111 1111</td><td>(285)</td><td>(End_Marker control)</td></tr>
</table>

38.10.2 ALDC_2 Algorithm - Formal definition (1024 byte history)

ALDC_2 algorithm is identical to ALDC_1 except for 10-bit displacement field

i.e., <displacement> := <b><b><b><b><b><b><b><b><b><b> (10-bit)

38.10.3 ALDC_4 Algorithm - Formal definition (2048 byte history)

ALDC_4 algorithm is identical to ALDC_1 except for 11-bit displacement field

i.e., <displacement> := <b><b><b><b><b><b><b><b><b><b><b> (11-bit)

Annex A
(informative)

No enhancements to FC-PH annex A.

Annex B
(informative)

No enhancements to FC-PH annex B.

Annex C
(informative)

Optical cable plant usage

In some cases, it will be desirable to use another multimode cable plant to those described in FC-PH 8.2 and 8.3. This may be due to the need for extended distances or for operation in locations where the cable plants are presently installed. These fibre types have not been studied and details for their use are not provided for the main body of this document. Therefore, using these fibre types may change the maximum achievable distance between nodes. See FC-PH annex D for methods of computing distances.

Table C.1 – Additional fibre types

Nomenclature	Primary Sub-clause	Fibre Type	Distance
400-M6-SN-I	6.2.4	62,5 µm	2m-90m
200-M6-SN-I	6.2.4	62,5 µm	2m-150m
100-M6-SN-I	6.2.4	62,5 µm	2m-300m

Annex D
(informative)

No enhancements to FC-PH annex D.

Annex E
(informative)

Optical cable plant usage

Table E.2 shows rows added to FC-PH Table E.2.

Table E.2 – Calculated max. single-mode length

FC-0	LB	N_{CON}	Length
400-SM-LL-I 200-SM-LL-I	8-2 = 6	4	2,1 km

E.5 Type SN multimode links

Type SN shortwave multimode links without OFC are designed to have a minimum link budget of 6 dB. Of this 6 dB, 4 dB is allocated to cable plant losses and 2 dB is allocated to system power penalties. To control the system power penalties to less than 2 dB, loss at a single connector has been limited to 0.56 dB with a maximum of 4 connectors.

Because of the maximum connector loss restriction, section E.4 in FC-PH, is no longer a valid method to determine maximum link length. To determine contribution of connector loss use the 0.56 dB maximum loss. The attenuation vs. length calculation is still valid.

The maximum length limit based on fiber bandwidth is approximately 500 meters for 50 um MMF and 300 meters for 62.5 um MMF.

Annex F
(informative)

No enhancements to FC-PH annex F.

Annex G
(informative)

No enhancements to FC-PH annex G.

Annex H
(informative)

No enhancements to FC-PH annex H.

Annex I
(informative)

No enhancements to FC-PH annex I.

Annex J
(informative)

System jitter allocation

Table J.1 shows rows added to FC-PH table J.1.

Table J.1 – Jitter allocations

Variant	t_r/t_f (ns)		Jitter (UI)							
	S	R	component	b	b to S	S	S to R	R	R to c	c
400-SM-LL-L	0,11	NA	DJ	0,08	0,12	**0,20**	0	0,20	0,08	0,28
			RJ	0,12	0,20	0,23	0	0,23	0,35	0,42
			Total	0,20	0,32	**0,43**	0	0,43	0,43	0,70
400-M5-SN-I	0,11	0,15	DJ	0,08	0,12	**0,20**	0,03	0,23	0,08	0,31
			RJ	0,12	0,20	0,23	0	0,23	0,31	0,39
			Total	0,20	0,32	**0,43**	0,03	0,46	0,39	0,70
200-SM-LL-L	0,22	NA	DJ	0,08	0,12	**0,20**	0	0,20	0,08	0,28
			RJ	0,12	0,20	0,23	0	0,23	0,35	0,42
			Total	0,20	0,32	**0,43**	0	0,43	0,43	0,70
200-M5-SN-I	0,22	0,3	DJ	0,08	0,12	**0,20**	0,03	0,23	0,08	0,31
			RJ	0,12	0,20	0,23	0	0,23	0,31	0,39
			Total	0,20	0,32	**0,43**	0,03	0,46	0,39	0,70
100-M5-SN-I	0,45	0,6	DJ	0,08	0,12	**0,20**	0,03	0,23	0,08	0,31
			RJ	0,12	0,20	0,23	0	0,23	0,31	0,39
			Total	0,20	0,32	**0,43**	0,03	0,46	0,39	0,70
25-M5-LE-I	**2,0/2,2**	2,5	DJ	0,08	0,08	**0,16**	0,03	**0,19**	0,09	0,28
			RJ	0,08	0,03	**0,09**	0	**0,09**	0,41	0,42
			Total	0,16	0,11	0,25	0,03	0,28	0,50	0,70

NOTE - For lasers and VCSELs the transition times are made at the 20% to 80% points when measured through a fourth order Bessel-Thompson filter.

Annex K
(informative)

No enhancements to FC-PH annex K.

Annex L
(informative)

No enhancements to FC-PH annex L.

Annex M
(informative)

No enhancements to FC-PH annex M.

Annex N
(informative)

No enhancements to FC-PH annex N.

Annex O
(informative)

No enhancements to FC-PH annex O.

Annex P
(informative)

No enhancements to FC-PH annex P.

Annex Q
(informative)

No enhancements to FC-PH annex Q.

Annex R
(informative)

No enhancements to FC-PH annex R.

Annex S
(informative)

No enhancements to FC-PH annex S.

Annex T
(informative)

No enhancements to FC-PH annex T.

Annex U
(informative)

No enhancements to FC-PH annex U.

Annex V
(informative)

No enhancements to FC-PH annex V.

Annex W
(informative)

FC-PH-2 bibliography

[1] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," IEEE Trans. Inform. Theory, vol. IT-23, no.3, pp. 337-343, 1977.

[2] J. A. Storer and T. G. Szymanski, "Data compression via textual substitution," J. ACM, vol. 29, no. 4, pp. 928-951, 1982.

[3] T. C. Bell, "Better OPM/L Text Compression," IEEE Trans. Comm. vol. COM-34, no. 12, December, 1986.

[4] R. P. Brent, "A Linear Algorithm for Data Compression," Australian Computer J., vol. 19, no. 2, pp. 64-68, May, 1987.

[5] ALDC Product information packet, available from IBM Corp., 1000 River Street, Dept. N97/Bldg 861-2, Essex Junction, VT 05452-4299.

Annex X
(informative)

Hunt Group Models

Hunt Group Models are included in this annex for information.

X.1 Model 1

Model 1 is depicted in figure X.1.

In model 1, common control performs only the task of notifying each N_Port of the HG_ID and its associated Hunt Group Login Service Parameters. The Hunt Group Identifier functions only as an Exchange Responder since Exchange Control (ESB, X_ID) is managed by each N_Port. When a Data frame which originates an Exchange with D_ID = K, the Fabric selects either K1, K2, or K3 for frame delivery.

All resources are managed by each N_Port. Exchange control and Sequence control are local to each N_Port. The Hunt Group Class Service Parameters shall specify that X_ID interlock is required.

X.1.1 Classes supported

Model 1 is applicable to Class 1 or 2 support. This model does not support Hunt Groups in Class 3.

X.1.2 Addressing protocol

The use of S_ID and D_ID values for HG_ID = K in model 1 is specified in table X.1.

Responder

In model 1 when any member of HG_ID (for example, K2) receives the Data frame (D_ID = K, S_ID = X) which initiates a new Exchange, it assigns an RX_ID known to its

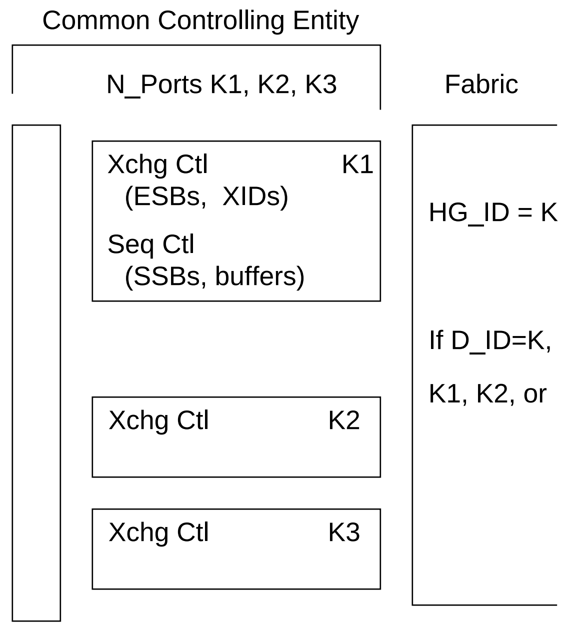


Figure X.1 – Model 1 - Hunt Group

N_Port and responds with an ACK frame with an S_ID of K2, and a D_ID of X.

From this point on, the Originator of the Exchange remains X and the Responder address identifier remains K2.

Originator In model 1 when any member of HG_ID = K originates an Exchange, it shall use its base address identifier of K1, K2, or K3 since all control elements are N_Port managed.

Table X.1 – Model 1 Addressing Protocol					
Originator / Responder	**S_ ID**	**D_ ID**	**CCE Consructs**	**N_Port Constructs**	**Comments**
Originator - all Data frames	Kn	X	None	ESB, OX_ID SSB	Kn as Originator uses its base Identifier of Kn on all frames of Exchange.
First Data frame as Responder	X	K	HG_ID, Common Service Parameters	ESB, RX_ID SSB	K(HG_ID) is D_ID of first Data frame of Exchange from X.
ACK to first frame as Responder	Kn	X	HG_ID, Common Service Parameters	ESB, RX_ID SSB	Kn reassigns S_ID in ACK to Kn since ESB and X_ID are managed by Kn. Exchange proceeds between Kn and X.
Responder - Other Data and ACK frames	Kn	X	HG_ID, Common Service Parameters	ESB, RX_ID SSB	Kn becomes D_ID of all frames from the Responder. The Exchange proceeds between Kn and X.

X.2 Model 2

Model 2 is depicted in figure X.2.

In model 2, common control provides resources associated with an Exchange (ESB, X_ID). The Hunt Group Identifier (HG_ID = K) functions as the S_ID and D_ID for Data and ACK frames throughout the Exchange when any member N_Port of HG_ID = K (K1, K2, or K3) is selected by the Fabric. Sequence control is retained by the N_Port.

X.2.1 Resource management

Exchange resources are managed CCE across the Hunt Group. Sequence resources (SSB) are managed by each member N_ Port.

X.2.2 Classes supported

Model 2 is only applicable to Class 1 support.

X.2.3 Addressing protocol

The use of S_ID and D_ID values for HG_ID = K in model 2 is specified in table X.2

Responder

In model 2 when any member of HG_ID (for example, K2) receives the Data frame (D_ID = K, S_ID = X) which initiates a new Ex change, it assigns an RX_ID known to the Hunt Group CCE and responds with an ACK frame with an S_ID of K, and a D_ID of X.

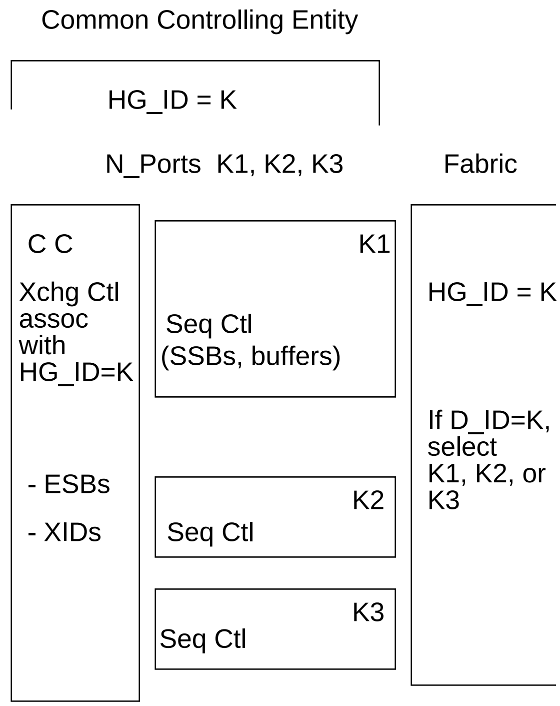


Figure X.2 – Model 2 - Hunt Group resources

Originator

In model 2 when any member of HG_ID = K originates an Exchange on behalf of the Hunt Group, it shall use HG_ID as the address identifier for all frames of the Exchange. As each Sequence is terminated Sequence Status information from the Sequence Status Block shall be used to update Exchange Status.

Table X.2 – Model 2 Addressing Protocol					
Originator / Responder	S_ID	D_ID	CCE Constructs	N_Port Constructs	Comments
Originator all Data frames	K	X	HG_ID, Common Service Parameters, ESB, OX_ID	SSB	Kn as Originator of Exchange uses HG_ID on first and all successive frames of Exchange
First Data frame as Responder	X	K	HG_ID, Common Service Parameters, ESB, RX_ID	SSB	K(HG_ID) is D_ID of first Data frame of Exchange from X.
ACK to first frame as Responder	K	X	HG_ID, Common Service Parameters, ESB, RX_ID	SSB	Kn retains S_ID in ACK of K since ESB and X_ID are CCE managed. Exchange proceeds between K and X.
Responder - Other Data and ACK frames	K	X	HG_ID, Common Service Parameters, ESB, RX_ID	SSB	K (HG_ID) is S_ID for all other frames of Exchange with X.

X.3 Model 3

Model 3 is depicted in figure X.3.

In model 3, common control provides resources associated with an Exchange (ESB, X_ID) as well as with each Sequence (SSB). The Hunt Group Identifier (HG_ID = K) functions as the S_ID and D_ID for Data and ACK frames throughout the Exchange when any member N_Port of HG_ID = K (K1, K2, or K3) is selected by the Fabric. Sequence control shall be shared across any member N_Port (Class 2, 3).

X.3.1 Resource management

Both Exchange and Sequence resources shall be CCE managed across the Hunt Group.

X.3.2 Classes supported

Model 3 is applicable to Class 1, 2 or 3 support.

X.3.3 Addressing protocol

The use of S_ID and D_ID values for HG_ID = K in model 3 is specified in table X.3.

Responder

In model 3 when any member of HG_ID (for example, K,) receives the Data frame S_ID = K, S_ID = X) which initiates a new Exchange, it assigns an RX_ID known to the Hunt Group CCE. and responds with an ACK frame with an S_ID of K, and a D_ID of X. A single member N_Port shall operate as Sequence Initiator for any given Sequence. Any member of the Hunt Group may receive frames as a Sequence Recipient in Class 2 or 3.

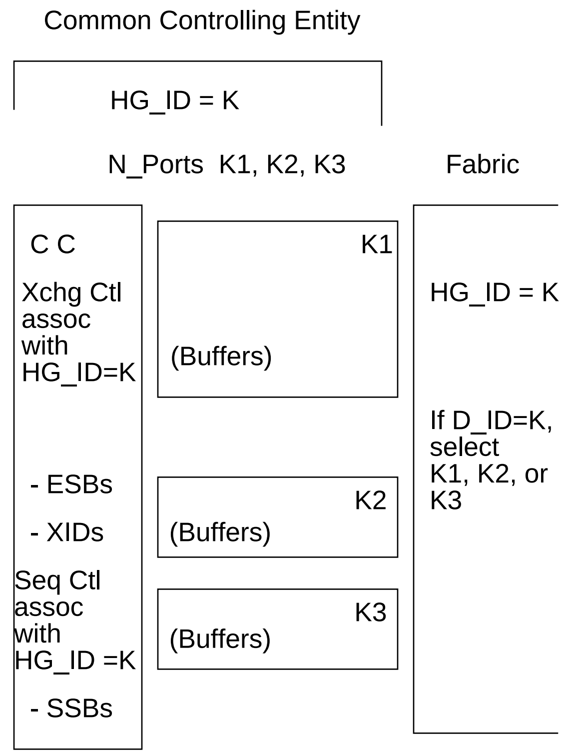


Figure X.3 – Model 3 - Hunt Group resources

Originator

In model 3 when any member of HG_ID = K originates an Exchange on behalf of the Hunt Group, it shall use HG_ID as the address identifier for all frames of the Exchange. A single member N_Port shall operate as Sequence Initiator for any given Sequence. Any member of the Hunt Group may receive frames as a Sequence Recipient in Class 2 or 3.

Table X.3 – Model 3 Addressing Protocol					
Originator / Responder	S_ID	D_ID	CCE Constructs	N_Port Constructs	Comments
Originator - all Data frames	K	X	HG_ID, Common Service Parameters, ESB, OX_ID SSB	Buffers	K as Originator of Exchange uses HG_ID on first and all successive frames of Exchange.
First Data frame as Responder	X	K	HG_ID, Common Service Parameters, ESB, RX_ID SSB	Buffers	K(HG_ID) is D_ID of first Data frame of Exchange from X.
ACK to first frame as Responder	K	X	HG_ID, Common Service Parameters, ESB, RX_ID SSB	Buffers	Kn retains S_ID in ACK of K since ESB, X_ID, and SSB are CCE managed. Exchange proceeds between K and X.
Responder - Other Data and ACK frames	K	X	HG_ID, Common Service Parameters, ESB, RX_ID SSB	Buffers	K (HG_ID) is S_ID for all other frames of Exchange with X.

X.4 Model 4

Model 4 is depicted in figure X.4.

X.4.1 Resource management

Both Exchange and Sequence resources In model 4, common control provides resources associated with an Exchange (ESB, X_ID) as well as with each Sequence (SSB) for use by each Hunt Group member. Hunt Group operation proceeds as in model 3.

In addition, each N_Port provides resources for Exchange and Sequence control when operating as Kn. This allows Kn to operate with Kn using its base identifier with its own Exchange and Sequence control elements as well as a member of the Hunt Group with HG_ID = K.

X.4.2 Resource management

Both Exchange and Sequence resources shall be CCE managed across the Hunt Group. In addition, each member N_Port may operate independently using its base identifier (Kn) with its Exchange and Sequence resources.The Hunt Group Class Service Parameters shall specify that X_ID interlock is required.

In model 4, a member N_Port may convert an Exchange directed to the HG_ID (CCE managed) of K to its base identifier Kn (N_Port managed) if common resources are unavailable. This is accomplished as a Responder by following theaddressing protocol of model 1 (see 4.7.3) or by using Kn (S_ID) as an Originator. In addition, the Hunt Group may be associated with limited, specific ULP TYPES while other ULP TYPES may be associated on a single N_Port basis.

X.4.3 Classes supported

Model 4 is applicable to Class 1, 2 or 3 support.

X.4.4 Addressing protocol

The use of S_ID and D_ID values for HG_ID = K in model 4 is specified in table 3 when operating as a Hunt Group member. Each N_Port Kn shall use Kn as S_ID and D_ID throughout Exchanges which are managed by that N_Port.

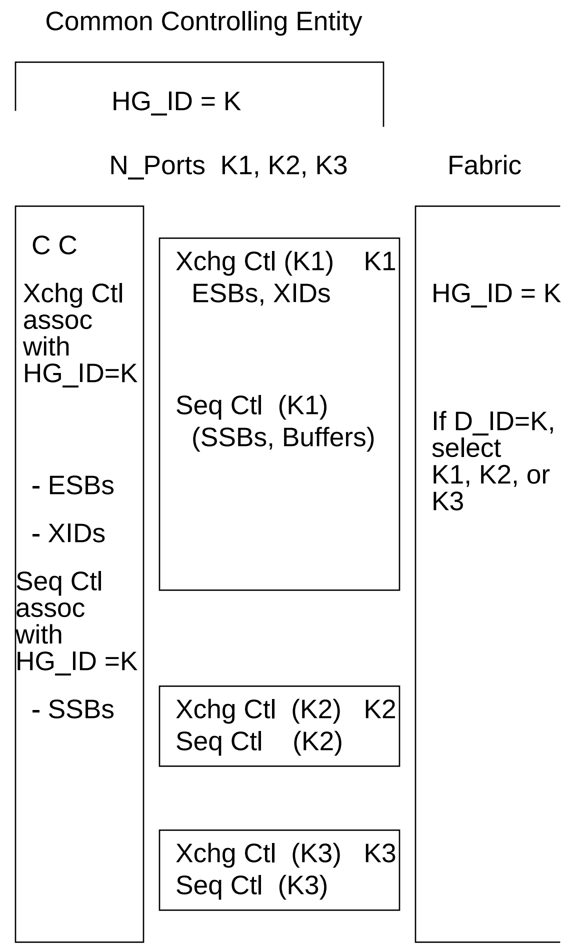


Figure X.4 – Model 4 - Hunt Group resources

Annex Y
((informative)

Available BB_Credit management example

The following is an example implementation using BB_Credit and available BB_Credit (see figure Y.1). Assume two Ports, A and B, and both are either N_Ports or L_Ports.

Port A has sixteen receive buffers available and a BB_Credit Login value of two for Port B. Port B has eight receive buffers available and a BB_Credit Login value of one for Port A.

If they are L_Ports, assume that L_Port A has arbitrated and won and is going to open B; full-duplex is used; and, both A and B have frames to transmit.

Port A:

1. looks up the BB_Credit Login value for Port B (two).
2. checks to see how many receive buffers it has available (sixteen).
3. transmits if in the Loop, OPN(B,A), two Fill Words and one R_RDY and three Fill Words. (If N_Port, step 3 is not applicable.)
4. transmits two frames (BB_Credit Login value for B). Note that had BB_Credit been zero (0), no frames could have been sent by Port A (prior to receiving R_RDY). The remaining fifteen R_RDYS are transmitted after the first frame.
5. receives and counts the R_RDYs sent by Port B (eight). Once Port A has received and discounted two R_RDYs (to account for the frames already shipped against Login BB_ Credit in step 4), Port A has an available BB_Credit of six that it may use to transmit up to six additional frames.
6. transmits one frame for each available BB_ Credit (and decrement the available BB_ Credit by one).
7. counts the number of R_RDYs received from Port B.
8. receives frames sent by Port B into the receive buffer(s).
9. transmits one R_RDY for each receive buffer that becomes available.
10. repeats steps 6 through 9 until all frames have been sent.
11. transmits CLS, if in the Loop.
12. continues to receive frames from B, but transmits no R_RDYs or frames.
13. receives CLS, if in the Loop and closes its end of the Loop.

Port B:

1. receives, if in Loop, OPN(B,A) and opens the Loop. (If N_Port, step 1 is not applicable.)
2. looks up the BB_Credit for Port A (one) and checks to see how many receive buffers it has available (eight).
3. transmits if in the Loop, two Fill Words and one R_RDY and three Fill Word. (If N_Port, step 4 is not applicable.)
4. transmits one frame (BB_Credit Login value for Port A). Note that had BB_Credit been zero (0), no frames could have been sent by B. The remaining seven R_RDYs are transmitted after the first frame.
5. receives and counts the number of R_RDYs sent by Port A. Once Port B has received and discounted one R_RDY (to account for the frame already shipped against BB_ Credit in step 5), Port B has an available BB_Credit of fifteen that it may use to transmit up to fifteen additional frames.
6. transmits one frame for each available BB_ Credit (and decrement the available BB_ Credit by one).
7. counts the number of R_RDYs received from Port A.
8. receives the number of frames sent by Port A into the receive buffer(s).
9. transmits an R_RDY for each receive buffer that becomes available.
10. may repeat steps 6 through 9 until the Available BB_Credit is exhausted.
11. transmits a CLS, if in the Loop.
12. Since the CLS has already been received, Port B closes its end of the Loop.

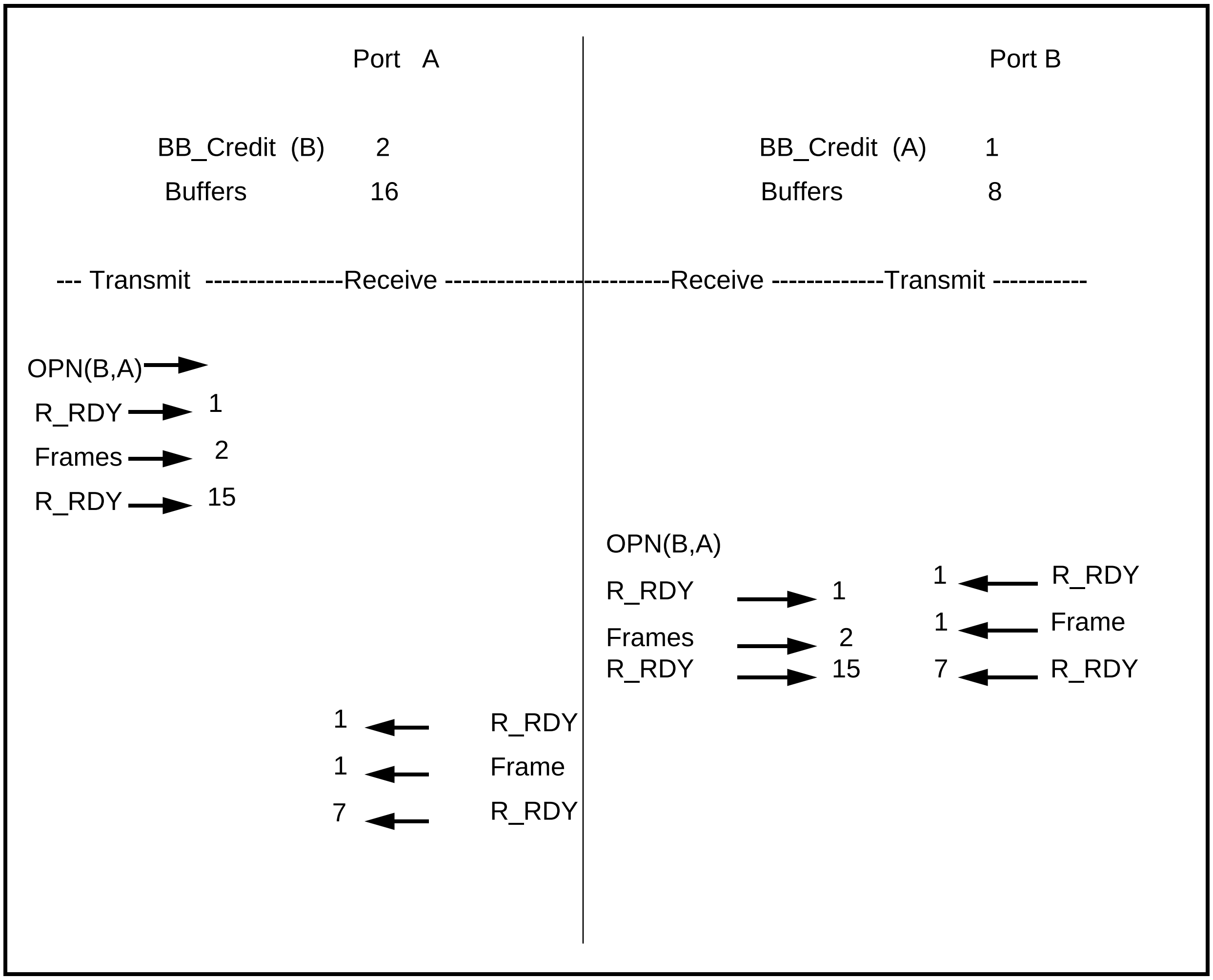


Figure Y.1 - Alternate_Credit Use Example

There are several variations on the previous example.

1. Data frame transfer is only from A to B even though OPN(B,A)(full-duplex) is used. Port A transmits one R_RDY followed by the number of frames represented by the BB_ Credit Login value. In this case, B has no Data frames to transmit, but transmits one R_RDY for each available receive buffer and FC-PH Link_Control frames (e.g., ACKs) to A. Note that B may limit the number of frames that A can transmit by transmitting one R_RDY for each available receive buffer, followed by CLS. Once Port A has received the CLS, it can then only transmit frames until its available BB_Credit is exhausted before it must close its end of the Loop.
2. BB_Credit is zero (0) on both ends. In this case, neither A nor B can transmit any frames until at least one R_RDY is received. For each R_RDY received, a frame may be sent. Note that when BB_Credit is zero (0), a Loop turn-around delay is required at A before transmitting the first (and possibly the only) frame. By guaranteeing a minimum of receive buffers (as indicated by BB_Credit), this turn-around delay may be eliminated.
3. Port A transmits an OPN(B,A) (full-duplex), followed by at least one R_RDY, followed by two frames (the BB_Credit Login value for Port B). Port B does not look up the BB_ Credit for A, and sends at least one R_RDY (one for each available receive buffer). Port B waits for the first R_RDY from A. When at least one R_RDY is received (i.e., available BB_Credit is one (1)), B can transmit one FC-PH frame.
4. Port A transmits an OPN(B,B) (half-duplex). In this case, Port B cannot identify A and must wait for the first frame from A to transmit any frames (note: since this is an half-duplex OPN, no data frames may be transmitted by B). Once a frame is received, the low-order byte of the S_ID is the AL_PA of A. B can then establish the BB_Credit for A. If B is using a minimum Loop value for BB_ Credit, the AL_PA of A is not required.
5. Port A transmits an OPN(B,B) (half-duplex). In this case, Port B must wait for the first R_ RDY from A. Once at least one R_RDY is received (i.e., available BB_Credit is one (1)), B can transmit one FC-PH Link Control frame to A.

Annex Z
(informative)

48-bit Global Identifier Format

Z.1 General

The IEEE defined 48-bit global identifier (GID_48) is assigned by a manufacturer who has been assigned a company_id value by the IEEE Registration Authority. The 48-bit identifier is a concatenation of the 24-bit company assigned by the IEEE Registration Authority and a 24-bit extension identifier assigned by the organization with that company_id assignment.

The IEEE administers the assignments of 24-bit company_id values. The assignments of these values are public, so that a user of a GID-48 value can identify the organization which provided that value. The IEEE/RAC has no control over the assignments of 24-bit extension identifiers and assumes no liability for assignments of duplicate GID-48 identifiers.

Z.2 Format

You may have a 48-bit global identifier (GID-48) provided by an authorized manufacturer of these values (which sell electronically readable chips). The most significant 24 bits of this value are the company_id value assigned to the manufacturer by the IEEE Registration Authority. The least-significant 24-bit extension identifier assigned by the manufacturer.

For example, assume that manufacturer's IEEE-assigned company_id value is $ACDE48_{16}$ and the manufacturer's extension identifier is 234567_{16}. The GID-64 value generated from these two numbers is $ACDE48234567_{16}$, whose byte and bit representations are illustrated in figure Z.1.

If provided in byte-addressable media, the original byte-address is specified: the most through least significant bytes of the GID-48 value are contained within the lowest through highest byte addresses, as illustrated.

When transferred to other standard-specific locations (within a disk file or network packet, for example) the relative ordering of the bytes may be changed, as specified within the applicable standard.

most-significant byte					least significant byte	
AC	DE	48	23	45	67	(hex)
1010 1100	1101 1110	0100 1000	0010 0011	0100 0101	0110 0111	(binary)
A+0	A+1	A+2	A+3	A+4	A+5	original byte address

Figure Z.1 – 48-bit global identifier example

Annex AA
(informative)

Fibre Channel use of IEEE Global Identifiers

The highest level architecture of Fibre Channel (FC) consists of multiple Nodes (workstations, disk arrays, supercomputers etc.) interconnected by an active intelligent interconnection scheme called a Fabric. Allowance is made for Nodes which have multiple ports (N_Ports), and each N_Port is connected to a separate Fabric port (F_Port). In the base FC standard - ANSI X3.230-1994, Fibre Channel Physical and Signaling Interface (FC-PH) - the N_Port to F_Port interface is defined explicitly but the operation of the Fabric is only defined in terms of a functional model. However other FC standards define Fabrics using specific technologies, and based on a specific topology.

The Fabric model defines a single, homogenous, 24 bit address space, and those "short" addresses are the basis for routing information within the Fabric. The protocol supported by the N_Port/F_Port interface is also capable of transporting "long" addresses of up to 60 bits in length. These addresses may be globally unique types defined by a number of standards bodies worldwide. FC-PH generically refers to these addresses as WorldWide Names (WWNs). Methods of address resolution are provided between the WWNs and the FC "short" addresses.

These WWNs have two different uses. First, they are used to provide unique identification of items within a FC configuration. Second, they are used to facilitate routing in situations where the FC configuration is a small part of a larger wide area network, or where the Upper Layer Protocols being transported by FC assume the use of those long addresses. A prime example of such an Upper Layer Protocol is the IEEE 802.2 Logical Link Layer protocols mapped to FC as defined by the FC Link Encapsulation (FC-LE) scheme.

For the first use:

a) It is acceptable that a single WWN be provided for each Fabric and each Node, and this is then used with a local qualifier to identify the specific port within the Node or Fabric;

b) It is also acceptable that a separate WWN be provided with each N_Port or F_Port.

The situation for the second use is more complex. Where:

a) an application designed for an IEEE 802 LAN is moved with minimal modification into a FC environment;

b) communication occurs between an FC-LE Entity and an IEEE 802 station on a LAN; or

c) FC provides an intermediate part of a path between two IEEE 802 stations on LANs (which is handled by IEEE 802.1G remote bridging);

it is strongly suggested that each N_Port and F-Port be assigned its own WWN.

In addition, the use of bridging techniques to communicate between Entities implementing FC-LE and systems on the IEEE 802 LAN to:

a) allow the FC-LE entity be able to address the system on the IEEE 802 LAN; and

b) allow the system on the IEEE 802 LAN to use ULAs to address the FC-LE Entity requires that each N_Port and F_Port be assigned its own WWN.

One of the formats defined as usable for a WWN is an IEEE 48 bit address format. This is specified to contain a 48 bit IEEE Standard 802.1A Universal LAN MAC Address (ULA). The ULA is represented as an ordered string of six bytes numbered from 0 to 5. The least significant bit of byte 0 is identified as the Individual/Group Address (I/G) bit. The next least significant bit is identified as the Universally or Locally Administered Address (U/L) bit. It is further specified that layout of the bytes in two FC standard 32 bit words is as follows:

Bits

63	55	47	39	32
		ULA Byte 0 UI / / LG	ULA byte 1	

ULA byte 2	ULA byte 3	ULA byte 4	ULA byte 5	
31	23	15	07	00

Bits

Note that all fields in the FC-PH standard are defined as having the most significant bit in the highest numbered bit position, and the most significant byte in the position closest to the start of the FC-PH Packet.

The FC-PH standard identifies no further details of the contents of the ULA.

Annex BB
(informative)

Class 4 Examples

Two ladder diagrams illustrating the operation of a single Class 4 circuit is shown in figures BB.1 and BB.2. Both figures include details on how End-to-end and VC credit are managed. Figure BB.1 shows how an N_Port request set-up of a Class 4 circuit, the subsequent circuit establishment, followed by an activation and deactivation cycle. Figure BB.2 shows the non-destructive collision of two activation requests, normal Class 4 frame and acknowledgment flow ending with the removal of the Class 4 circuit.

BB.1 Setup, activate and deactivate

The following describes in detail the Frame and Class 4 Primitive Signal flow shown in figure BB.1:

a) The Circuit Initiator, N_Port ‘A’, enters the Pending state and transmits a Quality of Service request (a) (QoSR) to the Quality of Service Facilitator (QoSF). The QoSF sets F_Port ‘A’ to the Pending state and builds two counter-directional virtual circuits across the Fabric, between F_Port ‘A’ and F_Port ‘B’, satisfying the QoS parameters.

b) The QoSF acknowledge (b) the receipt of the QoSR frame, if either Class 2 or 4 is used to request the Class 4 circuit setup.

c) F_Port ‘B’ enters the Pending state before the updated QoSR (c) frame is sent to N_port ‘B’. The QoSF determine which Service Class to use to propagate the setup request. N_Port ‘B’ enters the Pending state and initialize the service parameters, including the credit environment when it process the QoSR.

d) N_Port ‘B’ acknowledge (d) reception of the QoSR frame if the QoSF uses Class 2 or 4 to continue the Class 4 circuit setup process.

e) N_port ‘B’ transmits an ACC (e) frame, in response to the QoSR frame (c), once it is ready to proceed. The QoSF initialize the environment for the Class 4 circuit in response to the ACC frame.

f) The QoSF acknowledge (f) reception of the ACC if Class 2 or 4 was selected by the QoSF to complete setup of the Class 4 circuit.

g) When the Fabric environment have been initialized the QoSF return an ACC (g) frame to N_Port ‘A’ in response to the initial setup request (a). N_Port ‘A’ initialize the service parameters, including the credit environment when it process the ACC.

h) N_Port ‘A’ acknowledge (h) receipt of the ACC frame if N_Port ‘A’ selected Class 2 or 4 to request setup of the Class 4 circuit.

i) F_Port ‘A’ transmits a VC_RDY (i) and enters the Dormant state, when it is ready to accept an activation request. N_Port ‘A’ enters the Dormant state when it receives the VC_RDY and increases the VC credit limit and VC_Credit.

j) F_Port ‘B’ transmits a VC_RDY (j) and enters the Dormant state, when it is ready to accept an activation request. N_Port ‘B’ enters the Dormant state when it receives the VC_RDY and increases the VC credit limit and VC_Credit.

k) F_Port ‘A’ transmits a VC_RDY (k) advertising excess bandwidth. However, N_Port ‘A’ does not increase its VC_Credit when it receives the VC_RDY since VC_Credit already is at the VC_Credit limit.

l) N_Port ‘B’ transmits an activation request (l), consuming one EE_Credit and one VC_Credit. Transmission of the SOFc4 delimited causes N_Port to enter the Live state. This also change the VC credit limit to 2, the Live VC credit limit. F_Port ‘B’ enters the Live state when it receives the SOFc4 delimited activation frame. The frame then traverse the Fabric and is forwarded by F_Port ‘A’. F_Port ‘A’ enters the Live state when it transmits the SOFc4 delimiter. N_port 'A' enters the Live state when it receives the activation frame. This also changes the VC credit limit to the Live VC credit limit.

m) F_Port 'B' transmit a VC_RDY (m) to replenish VC credit. N_Port 'B' increases VC_Credit when it receives the VC_RDY.

n) F_Port 'B' transmit a VC_RDY (n) to replenish VC credit. N_Port 'B' increases VC_Credit when it receives the VC_RDY.

o) F_Port 'A' transmit a VC_RDY (o) to replenish VC credit. N_Port 'A' increases VC_Credit when it receives the VC_RDY.

p) F_Port 'A' transmit a VC_RDY (p) to replenish VC credit. However, N_Port 'A' does not increase VC_Credit since VC_Credit already is at the VC credit limit.

q) N_Port 'B' transmits a Data frame (q) consuming one EE_Credit and one VC_Credit. The frame is received by F_Port 'B', traverses the Fabric, is forwarded by F_Port 'A' and is received by N_Port 'A'. N_Port 'A' later acknowledge (w) the frame reception.

r) N_port 'B' transmits its last Data frame (r) to the Sequence initiated by (l) requesting deactivation of the Class 4 circuit. This consumes one EE_Credit and one VC_Credit. The frame is received by F_Port 'B', traverses the Fabric, is forwarded by F_Port 'A' and is received by N_Port 'A'. N_Port 'A' later acknowledge (x) and honor the deactivation request.

s) N_Port 'A' acknowledge reception of the activation frame (l) by transmitting an ACK frame (s), consuming one VC_Credit. The ACK frame is received by F_Port 'A', traverse the Fabric, is forwarded by F_Port 'B' and is received by N_Port 'B'. N_port 'B' reclaim one EE_Credit on receipt of the ACK frame.

t) F_Port 'A' transmit a VC_RDY (f) to replenish VC credit. N_Port 'A' increases VC_Credit when it receives the VC_RDY.

u) F_Port 'B' transmit a VC_RDY (u) to replenish VC credit. N_Port 'B' increases VC_Credit when it receives the VC_RDY.

v) F_Port 'B' transmit a VC_RDY (v) to replenish VC credit. N_Port 'B' increases VC_Credit when it receives the VC_RDY.

w) N_Port 'A' acknowledge reception of the Data frame (q) by transmitting an ACK frame (w), consuming one VC_Credit. The ACK frame is received by F_Port 'A', traverse the Fabric, is forwarded by F_Port 'B' and is received by N_Port 'B'. N_port 'B' reclaim one EE_Credit on receipt of the ACK frame.

x) N_Port 'A' acknowledge reception of Data frame (r) by transmitting an ACK frame (x), consuming one VC_Credit. N_Port 'A' elects to honor the deactivation request made by N_Port 'B'. The EOFdt delimited ACK frame is received by F_Port 'A', traverses the Fabric, is forwarded by F_Port 'B' and is received by N_Port 'B'. N_Port 'A' enters the Dormant state when it transmits the EOFdt delimiter, F_Port 'A' enters the Dormant state when it receives the EOFdt delimiter, F_Port 'B' enters the Dormant state when it transmit the EOFdt delimiter and N_Port 'B' enters the Dormant state when it receives the EOFdt delimiter. N_Port 'B' reinitializes EE_Credit on receipt of the ACK frame since the Class 4 virtual connection is now Dormant.

y) F_Port 'B' transmit a VC_RDY (y) to replenish VC credit. However, N_Port 'B' does not increase VC_Credit since VC_Credit already is at the VC credit limit.

z) F_Port 'A' transmit a VC_RDY (z) to replenish VC credit. N_Port 'A' increases VC_Credit when it receives the VC_RDY.

BB.2 Activation collision and removal

The following describes in detail the Frame and Class 4 Primitive Signal flow shown in figure BB.2:

a) N_Port 'A' transmits an activation request (a), consuming one EE_Credit and one VC_Credit, causing N_Port 'A' to enter the Live state from the Dormant sate. This also changes the VC_Credit limit to the Live VC credit limit. F_Port 'A enters the Live state when it receives the SOFc4 delimited activation frame. The frame then traverse the Fabric and is forwarded by F_Port 'B'. However, F_Port 'B' is already in the Live state, due to the reception of an SOFc4 delimited frame (b) transmitted by N_Port ‘B’. N_Port ‘B’ receives the SOFc4 delimited activation frame and later acknowledge it (m).

b) N_Port ‘B’ consumes one EE_Credit and one VC_Credit when it transmits the activation frame (b). Transmitting the activation frame causes N_Port ‘B’ to enter the Live state and increases the VC credit limit. F_Port ‘B’ enters the Live state when it receives the activation frame. This frame then transverse the Fabric and is forwarded by F_Port ‘A’ to N_Port ‘A’. N_Port ‘A’ later acknowledge (j) reception of the SOFc4 delimited activation frame. Both F_Port ‘A’ and N_Port ‘A’ remain in the Live state.

c) F_Port ‘A’ transmits a VC_RDY (c) to replenish VC credit. N_Port ‘A’ increases VC_Credit when it receives the VC_RDY.

d) F_Port ‘B’ transmits a VC_RDY (d) to replenish VC credit. N_Port ‘B’ increases VC_Credit when it receives the VC_RDY.

e) F_Port ‘A’ transmits a VC_RDY (e) to replenish VC credit. N_Port ‘A’ increases VC_Credit when it receives the VC_RDY.

f) F_Port ‘B’ transmits a VC_RDY (f) to replenish VC credit. N_Port ‘B’ increases VC_Credit when it receives the VC_RDY.

g) N_Port ‘A’ transmits its last Data frame (g) requesting removal of the Class 4 circuit, consuming one EE_Credit and one VC_Credit. The frame is received by F_Port ‘A’, traverses the Fabric, is forwarded by F_Port ‘B’ and is received by N_Port ‘B’. N_Port ‘B’ later honor the removal request when it acknowledge (r) the Data frame.

h) N_Port ‘B’ transmits its last Data frame (h) to the Sequence initiated by (b). This consumes one EE_credit and one VC_Credit. The frame is received by F_Port ‘B’, traverses the Fabric, and is forwarded by F_Port ‘A’ before it is received by N_Port ‘A’. N_Port ‘A’ later acknowledge the frame reception (l).

i) F_Port ‘B’ transmits a VC_RDY (i) to replenish VC credit. N_Port ‘B’ increases VC_Credit when it receives the VC_RDY.

j) N_Port ‘A’ acknowledge reception of Data frame (b) by transmitting an ACK frame (j), consuming one VC_Credit. The ACK frame is received by F_Port ‘A’, traverses the Fabric, is forwarded by F_Port ‘B’ and is received by N_Port ‘B’. N_Port ‘B’ reclaim EE_Credit on receipt of the ACK frame.

k) F_Port ‘A’ transmits a VC_RDY (k) to replenish VC credit. N_Port ‘A’ increases VC_Credit when it receives the VC_RDY.

l) N_Port ‘A’ acknowledge reception of Data frame (h) by transmitting an ACK frame (l), consuming one VC_Credit. The ACK frame is received by F_Port ‘A’, traverses the Fabric, is forwarded by F_Port ‘B’ and is received by N_Port ‘B’. N_Port ‘B’ reclaim EE_Credit on receipt of the ACK frame.

m) N_Port ‘B’ acknowledge reception of Data frame (a) by transmitting an ACK frame (m), consuming one VC_Credit. The ACK frame is received by F_Port ‘B’, traverses the Fabric, is forwarded by F_Port ‘A’ and is received by N_Port ‘A’. N_Port ‘A’ reclaim EE_Credit on receipt of the ACK frame.

n) F_Port ‘A’ transmits a VC_RDY (n) to replenish VC credit. N_Port ‘A’ increases VC_Credit when it receives the VC_RDY.

o) F_Port ‘A’ transmits a VC_RDY (o) to replenish VC credit. N_Port ‘A’ increases VC_Credit when it receives the VC_RDY.

p) F_Port ‘B’ transmits a VC_RDY (p) to replenish VC credit. N_Port ‘B’ increases VC_Credit when it receives the VC_RDY.

q) F_Port ‘B’ transmits a VC_RDY (q) to replenish VC credit. However, N_Port ‘B’ does not increase its VC_Credit when it receives the VC_RDY since VC_Credit is already at the VC credit limit.

r) N_Port ‘B’ acknowledge reception of Data frame (g) by transmitting an ACK frame (r), consuming one VC_Credit. N_Port ‘B’ elects to honor the removal request made by N_Port ‘A’. The EOFrt delimited ACK frame is received by F_Port ‘B’, traverses the Fabric, is forwarded by F_Port ‘A’ and is received by N_Port ‘A’. N_Port ‘B’ enters the Nonexisting state when it transmits the EOFrt delimiter, F_Port ‘B’ enters the Nonexisting state when it receives the EOFrt delimiter, F_Port ‘A’ enters the Nonexisting state when it transmit the EOFrt delimiter and N_Port ‘A’ enters the Nonexisting state when it receives the EOFrt delimiter. N_Port ‘A’ do not reclaim EE_Credit on receipt of the ACK frame since the Class 4 virtual connection is now nonexistent. However, N_Port ‘A’ final Sequence still completes successfully.

s) F_Port ‘A’ transmits a VC_RDY (s) to replenish VC credit. However, N_Port ‘A’ does not increase its VC_Credit when it receives the VC_RDY since VC_Credit is already at the VC credit limit.

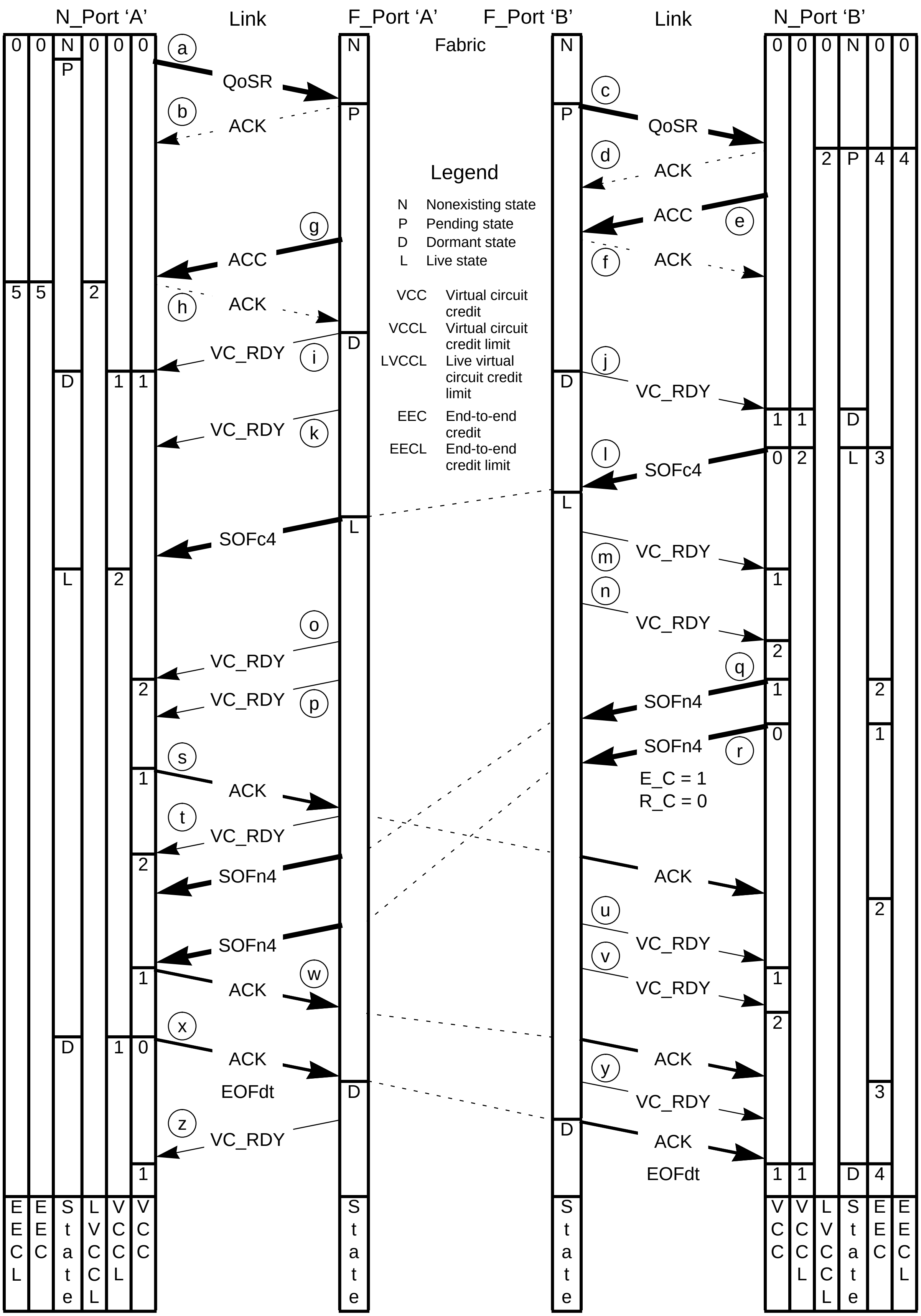


Figure BB.1 – Class 4 setup, activation and deactivation

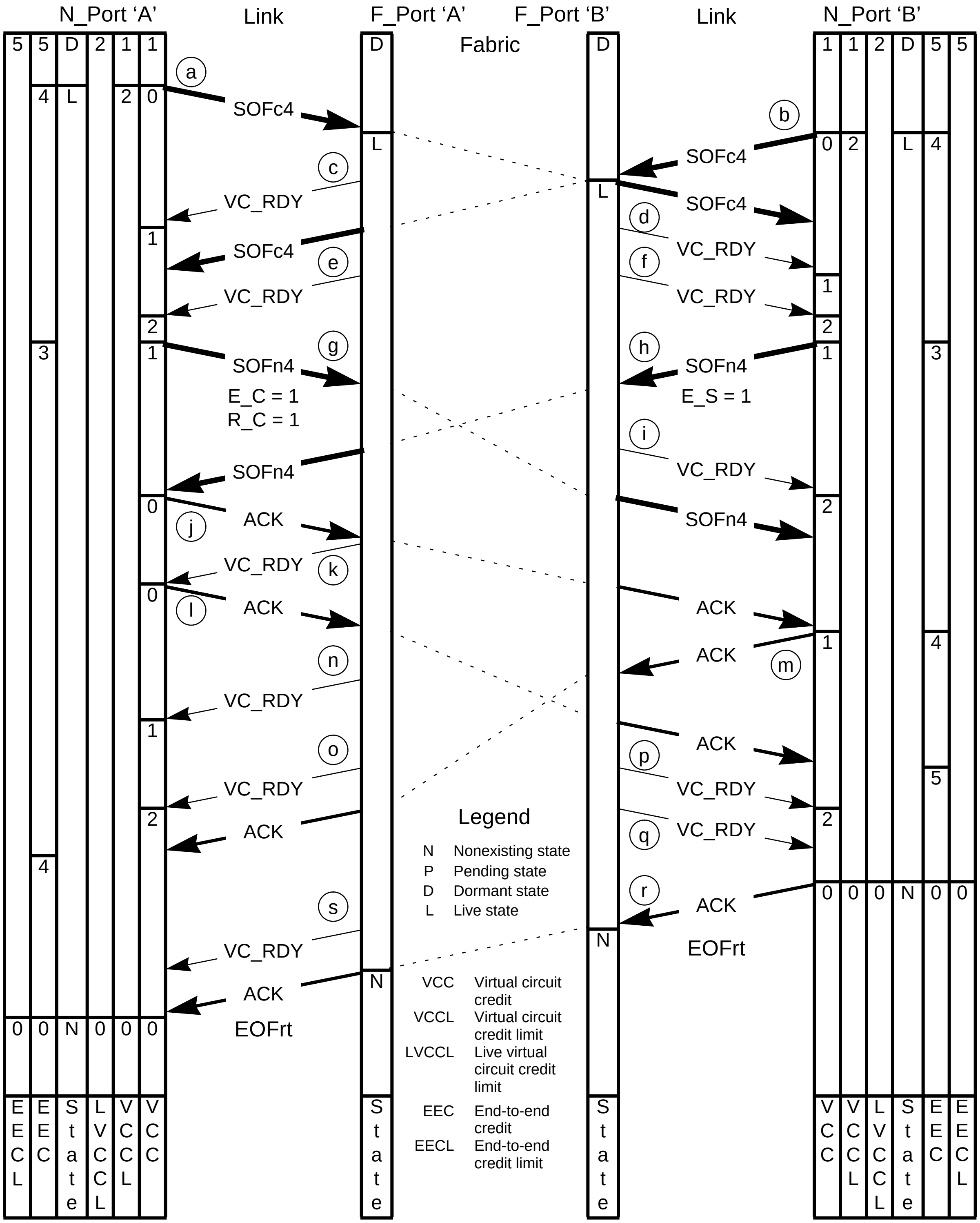


Figure BB.2 – Class 4 activation collision and circuit removal

Index

Index

Index

End of

Document