INTERNATIONAL STANDARD

ISO/IEC 14165-226

First edition
2020-01

Information technology — Fibre channel —

Part 226:
Single-byte command code sets mapping protocol - 6 (FC-SB-6)

Reference number
ISO/IEC 14165-226:2020(E)

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see http://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

This document was prepared by INCITS (as INCITS 544-2018) and drafted in accordance with its editorial rules. It was assigned to Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 25, *Interconnection of information technology equipment*, and adopted under the "fast-track procedure".

A list of all parts in the ISO/IEC 14165 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Table of Contents

List of Figures

List of Tables

Foreword

(This foreword is not part of American National Standard INCITS 544-2018.)

The Fibre Channel Single-Byte Command Code Sets - 6 (FC-SB-6) standard describes the Fibre Channel mapping protocol associated with the Single-Byte Command Code Sets.

This standard was developed by the INCITS Fibre Channel T11 Technical Committee (FC-TC) of Accredited Standards Committee during 2015-2016. The standards approval process started in 2016. 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 InterNational Committee for Information Technology Standards (INCITS), ITI, 1101 K Street, NW, Suite 610, Washington, DC 20005.

This standard was processed and approved for submittal to ANSI by INCITS. Committee approval of this standard does not necessarily imply that all committee members voted for its approval. At the time it approved this standard, INCITS had the following members:

Laura Lindsay, Chair
Donald Deutsch, Vice-Chair

Organization Represented	*Name of Representative*
Adobe Systems, Inc.	Scott Foshee
AIM Global, Inc.	Steve Halliday
	Mary Lou Bosco (Alt.)
	Chuck Evanhoe (Alt.)
Apple	Helene Workman
	Virginia Fournier (Alt.)
	David Singer (Alt.)
CA Technologies	Joel Fleck
	Paul Lipton (Alt.)
Department of Commerce - NIST	Michael Hogan
	Wo Chang (Alt.)
	Sal Francomacaro (Alt.)
Farance, Inc.	Frank Farance
	Timothy Schoechle (Alt.)
Futurewei Technologies, Inc.	Yi Zhao
	Wilbert Adams (Alt.)
	Wael Diab (Alt.)
	Timothy Jeffries (Alt.)
Google	Catherine Nelson
	Jutta Williams (Alt.)
GS1GO	Charles Biss
	Andrew Hearn (Alt.)
	Edward Merrill (Alt.)
HP, Inc.	Karen Higginbottom
	Paul Jeran (Alt.)
IBM Corporation	Steve Holbrook
	Alexander Tarpinian (Alt.)
Intel Corporation	Philip Wennblom
	Grace Wei (Alt.)
Microsoft Corporation	Laura Lindsay
	John Calhoon (Alt.)

Organization Represented	*Name of Representative*
Oracle	Donald Deutsch
	Anish Karmarkar (Alt.)
	Michael Kavanaugh (Alt.)
	Peter Lord (Alt.)
	Jim Melton (Alt.)
	Jan-Eike Michels (Alt.)
	Elaine Newton (Alt.)
	Toshihiro Suzuki (Alt.)
Telecommunications Industry Association (TIA)	Florence Otieno
United States Department of Defense	Angelo Curcio
	Sherman Charles (Alt.)
	Laura Hart (Alt.)
	Son (David) Kye (Alt.)
	Johnnie Mah (Alt.)
	Michael Southard (Alt.)
VMware, Inc.	Stephen Diamond
	Salim AbiEzzi (Alt.)
	Lawrence Lamers (Alt.)

Technical Committee T11 on Device-Level I/O Interfaces, which reviewed this standard, had the following members:

Steve Wilson, Chair
Claudio DeSanti, Vice-Chair
Richard Johnson, Secretary

Organization Represented	*Name of Representative*
Amphenol Corporation	Gregory McSorley
	Chris Lyon (Alt.)
	Alex Persaud (Alt.)
	Michael Wingard (Alt.)
	Daniel Dillow (Alt.)
	Brad Brubaker (Alt.)
	Mike Davis (Alt.)
	Dave Sideck (Alt.)
Anritsu Corporation	Ken Mochizuki
	Harry Pearson (Alt.)
Broadcom Technologies	Gautam Shiroor
	David Baldwin (Alt.)
	Evan Beauprez (Alt.)
	Adam Healey (Alt.)
	James Smart (Alt.)
	Ben Chu (Alt.)
	Mark Jones (Alt.)
Brocade	Steven Wilson
	John Crandall (Alt.)
	Scott Kipp (Alt.)
	David Peterson (Alt.)
CIENA	Stephen Shew
	Sebastien Gareau (Alt.)
Cisco Systems, Inc.	Claudio DeSanti
	Fabio Maino (Alt.)
	Landon Noll (Alt.)
	Joe Pellisier (Alt.)
	Federico Scandroglio (Alt.)

Organization Represented	*Name of Representative*
CommScope	Richard Baca
	Richard Case (Alt.)
	Jack Jewell (Alt.)
	Paul Kolesar (Alt.)
	Joe Livingston (Alt.)
	Yang Xu (Alt.)
Corning, Inc.	Doug Coleman
	Steven Swanson (Alt.)
Data Center Systems	Kevin Ehringer
	John Amato (Alt.)
	Todd Wheeler (Alt.)
Dell, Inc.	Joseph White
	Gaurav Chawla (Alt.)
	Jeff Young (Alt.)
EMC Corporation	Louis Ricci
	David Black (Alt.)
	Erik Smith (Alt.)
Finisar Corporation	Chris Yien
	Richard Johnson (Alt.)
Foxconn Interconnect Technology, Ltd.	John Petrilla
	Randy Clark (Alt.)
	Fred Fons (Alt.)
	Glenn Moore (Alt.)
Fujitsu America, Inc.	Mark Malcolm
	Kun Katsumata (Alt.)
	Osamu Kimura (Alt.)
	Gene Owens (Alt.)
Futurewei Technologies, Inc.	Hao Chen
	Xiaoyu Ge (Alt.)
	Jincheng Li (Alt.)
	Jia Shi (Alt.)
	Wei Song (Alt.)
GlobalFoundries US2 LLC	Adrian Butter
Hewlett-Packard Enterprise Corporation	Barry Maskas
	Siamack Ayandeh (Alt.)
	Rupin Mohan (Alt.)
	Krishna Puttagunta (Alt.)
Hitachi America, Ltd.	Hideo Saito
	Akira Deguchi (Alt.)
	Akio Nakajima (Alt.)
Hitachi Data Systems	Eric Hibbard
	Vincent Franceschini (Alt.)
	Michael Hay (Alt.)
	Gary Pilafas (Alt.)
	Akinobu Shimada (Alt.)
	Ken Wood (Alt.)
IBM Corporation	Roger Hathorn
	Adrian Butter (Alt.)
	Patty Driever (Alt.)
	John Ewen (Alt.)
Keysight Technologies, Inc.	Joachim Vobis
	Stephen Didde (Alt.)
	Steve Sekel (Alt.)
Lumentum Operations	David Lewis
Luxtera	Tom Palkert
Microsoft Corporation	Lee Prewitt
	Paul Luber (Alt.)
	Steve Olsson (Alt.)
Molex, Inc.	Jay Neer
	Mark Bugg (Alt.)
	Scott Sommers (Alt.)
NetApp, Inc.	Frederick Knight
Oclaro, Inc.	Koichi Tamura
OFS Optics	Roman Shubochkin
Optoway	PC Chen
	Judy Chen (Alt.)

Organization Represented	*Name of Representative*
Oracle	Roger Dickerson
	Matt Gaffney (Alt.)
	Hyon Kim (Alt.)
	Doug Meyers (Alt.)
	Michael Roy (Alt.)
Panduit Corporation	Robert Elliott
	Jose Castro (Alt.)
	Robert Reid (Alt.)
	Steve Skiest (Alt.)
QLogic Corporation	Craig Carlson
	Michael Dudeck (Alt.)
	Alan Spalding (Alt.)
	Dean Wallace (Alt.)
Semtech Canada Corporation	David Brown
	Ed Frian (Alt.)
	Francois Tremblay (Alt.)
TE Connectivity	Nathan Tracy
	Melissa Knox (Alt.)
	Jeffery Mason (Alt.)
	Andy Nowak (Alt.)
The University of New Hampshire Interoperabilty Laboratory	Timothy Sheehan
Unisys Corporation	Jeffrey Dremann
	Phil Shelton (Alt.)
Viavi Solutions, Inc.	Jason Rusch
	Scott Baxter (Alt.)
	George Bullis (Alt.)
	Paul Gentieu (Alt.)
VMware, Inc.	Neil MacLean
	Murali Rajagopal (Alt.)
	Ahmad Tawil (Alt.)

Members Emeritus

James Coomes
William Ham
Robert Kembel
Joe Mathis
Gary Stephens
Horst Truestedt
Schelto Van Doorn

Task Group T11.3 on Fibre Channel Interconnection Schemes, which developed and reviewed this standard, had the following members:

Craig Carlson, Chair
Lou Ricci, Vice-Chair
Patty Driever, Secretary

Organization Represented	*Name of Representative*
Avago Technologies	Gautam Shiroor
	David Baldwin (Alt.)
	Adam Healey (Alt.)
	James Smart (Alt.)
Broadcom Limited	Pat Thaler
	Ben Chu (Alt.)
	Kumaran Krishnasamy (Alt.)
Brocade	David Peterson
	John Crandall (Alt.)
	Howard Johnson (Alt.)
	Steven Wilson (Alt.)

Organization Represented	*Name of Representative*
Cisco Systems, Inc.	Mike Blair
	Claudio DeSanti (Alt.)
	Fabio Maino (Alt.)
	J. Metz (Alt.)
	Landon Noll (Alt.)
	Joe Pelissier (Alt.)
Dell, Inc.	Louis Ricci
	David Black (Alt.)
	Gaurav Chawla (Alt.)
	Erik Smith (Alt.)
	Joseph White (Alt.)
	Jeff Young (Alt.)
Fujitsu America, Inc	Mark Malcolm
	Kun Katsumata (Alt.)
	Gene Owens (Alt.)
GlobalFoundries US2 LLC	Adrian Butter
	John Ewen (Alt.)
Hewlett Packard Enterprise	Barry Maskas
	Siamack Ayandeh (Alt.)
	Rupin Mohan (Alt.)
	Krishna Puttagunta (Alt.)
Hitachi Data Systems	Eric Hibbard
	Vincent Franceschini (Alt.)
	Michael Hay (Alt.)
IBM Corporation	Roger Hathorn
	Patty Driever (Alt.)
Microsoft Corporation	Lee Prewitt
	Paul Luber (Alt.)
NetApp, Inc.	Frederick Knight
	Urmi Misra (Alt.)
Oracle	Roger Dickerson
	Joe Chen (Alt.)
	Matt Gaffney (Alt.)
	Hyon Kim (Alt.)
	Doug Meyers (Alt.)
	Michael Roy (Alt.)
QLogic Corporation	Craig Carlson
	Raul Oteyza (Alt.)
	Darren Trapp (Alt.)
	Dean Wallace (Alt.)
Teledyne LeCroy Corporation	David Rodgers
Unisys Corporation	Jeffrey Dremann
	Phil Shelton (Alt.)
Viavi Solutions, Inc.	Jason Rusch
	Scott Baxter (Alt.)
	George Bullis (Alt.)
	Paul Gentieu (Alt.)
VMware, Inc.	Neil MacLean
	Murali Rajagopal (Alt.)
	Ahmad Tawil (Alt.)

Members Emeritus

James Coomes
William Ham
Robert Kembel
Joe Mathis
Gary Stephens

Introduction

FC-SB-6 describes the Fibre Channel protocol mapping for the Single-Byte Command Code Sets. FC-SB-6 is one of a number of Fibre Channel protocol mappings, referred to as FC-4s.

Fibre Channel –
Single-Byte Command Code Sets
Mapping Protocol - 6
(FC-SB-6)

1 Scope

This standard describes a communication interface between a channel and I/O control units that utilize the Single-Byte Command Code Sets (SBCCS) as implemented in a wide range of data processing systems. It employs information formats and signaling protocols that provide a uniform means for communicating with various types of I/O control units, facilitating a high bandwidth, high performance, and long distance information exchange environment. The signaling protocols and information exchanges are defined at a layer (FC-4) to compatibly utilize the link services and other functions provided by the INCITS Fibre Channel Framing and Signaling (FC-FS-4) and the INCITS Fibre Channel Link Services (FC-LS-3) specifications. This FC-4 Upper Level Protocol is referred to as the Fibre Channel-Single-Byte-6 Command Code Sets Mapping Protocol (FC-SB-6).

This standard modifies the FC-SB-5 standard to specify enhancements and clarifications to the command-mode and transport-mode protocols to increase the efficiency and expand the capabilities of operations.

.

2 Normative References

2.1 Qualification and availability of references

The references listed in this clause contain provisions that, through reference in this text, constitute provisions of this document. 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 in this clause.

Orders for ISO Standards and ISO publications should normally be addressed to the ISO member in your country. If that is impractical, ISO Standards and ISO publications may be ordered from ISO Central Secretariat (ISO/CS):

Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
E-mail: customerservice@iso.org
Post: ISO, 8, chemin de Blandonnet,
CP 401 - 1214 Vernier, Geneva - Switzerland

In order to avoid delivery errors, it is important that you accurately quote the standard's reference number given in the ISO catalogue. For standards published in several parts, you should specify the number(s) of the required part(s). If not, all parts of the standard will be provided.

Copies of the following documents may be obtained from ANSI, an ISO member organization:

Approved ANSI standards;
approved international and regional standards (ISO and IEC); and
approved foreign standards (including JIS and DIN).

For further information, contact the ANSI Customer Service Department:

Phone: +1 212-642-4980
Fax: +1 212-302-1286
Web: http://www.ansi.org
E-mail: ansionline@ansi.org

or the InterNational Committee for Information Technology Standards (INCITS):

Phone: 202-626-5737
Web: http://www.incits.org
E-mail: incits@itic.org

IETF Request for Comments (RFCs) may be obtained directly from the IETF web site at http://www.ietf.org/rfc.html.

2.2 Approved References

FC-PI-2:INCITS 404-2006, *Fibre Channel Physical Interfaces - 2 (FC-PI-2)*

FC-PI-5:INCITS 479-2011, *Fibre Channel Physical Interfaces - 5 (FC-PI-5)*

FC-PI-6:INCITS 512-2015, *Fibre Channel - Physical Interface - 6 (FC-PI-6)*

FC-PI-6P:INCITS 533-201x, *Fibre Channel - Physical Interface - 6 (FC-PI-6P)*

SBCON:ANSI X3.296-1997, *Single-Byte Command Code Sets Connection Architecture (SBCON)*

FC-SB-5:INCITS 485-2014, *Fibre Channel - Single Byte Command Code Sets - 5 (FC-SB-5)*

FCP-4:INCITS 481-2012, *Fibre Channel - Protocol - 4 (FCP-4)*

FDDI-MAC:ISO/IEC 9314-2:1989, *Fibre Distributed Data Interface - Media Access Control (FDDI-MAC)*

2.3 References Under Development

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

FC-FS-4:INCITS Project 2238-D, *Fibre Channel - Framing and Signaling - 4 (FC-FS-4)*

FC-LS-3:INCITS Project 2237-D, *Fibre Channel - Link Services - 3 (FC-LS-3)*

FC-SW-6:INCITS Project 2220-D, *Fibre Channel - Switch Fabric - 6 (FC-SW-6)*

FC-PI-7:INCITS 543-201X, *Fibre Channel - Physical Interface - 6 (FC-PI-7)*

3 Definitions and Conventions

3.1 Overview

For this standard, the following definitions and conventions apply.

3.2 Definitions

3.2.1 channel

entity, typically of a host computer, which consists of one N_Port and elements which performs the functions specified by this standard to provide access to I/O devices by means of control units or emulated control units

3.2.2 channel-command word (CCW)

a) control block which contains a request to perform an I/O operation in command mode; or b) structure of a specific system architecture which specifies the command to be executed along with parameters

3.2.3 CCW channel program

single channel-command word or sequence of channel-command words executed sequentially that control a specific sequence of channel operations

3.2.4 CCW I/O operation

decoding, accepting, and executing a CCW by an I/O device

3.2.5 channel image

single ULP instance of a channel having the logical appearance of a channel

3.2.6 channel-path identifier (CHPID*)*

system-unique 8-bit value assigned to each installed channel path of a system

3.2.7 channel program

a) CCW channel program; or b) TCW channel program

3.2.8 channel-to-channel connection

association between two channels, one of which provides an emulated control unit, which allows those two channels to exchange IUs on an exchange pair

3.2.9 command mode

mode of operation that is used to perform SB-6 device-level functions that are performed using an exchange pair and to perform all FC-SB-6 link-control functions

3.2.10 connection

association between a channel and control unit established: a) after the successful transfer of IUs that constitute an exchange pair resulting in two open exchanges, one inbound and the other outbound, and both occurring between the channel and the control unit in command mode; or b) after transfer of an IU by a channel to a control unit that opens an exchange in transport mode

3.2.11 control unit

physical or emulated entity, consisting of at least one N_Port and elements which adapt the characteristics of one or more I/O devices to allow their attachment to the N_Port of a channel

3.2.12 control-unit image

single ULP instance of a control unit having the logical appearance of a control unit

3.2.13 CRC Offset Block

list of 1-word values identifying the byte offset of each intermediate CRC word in the write transport data

3.2.14 device

equipment such as a printer, magnetic tape unit or direct-access-storage device (DASD) that is accessed by means of a control unit to allow attachment to a channel

3.2.15 device-command word (DCW)

a) control block which contains a device I/O command; or b) structure of a specific system architecture which specifies an I/O command to be executed along with parameters

3.2.16 device-level functions

protocols and functions used to perform and manage I/O operations

3.2.17 device information block (DIB)

data block present in all device-level FC-SB-6 command-mode IUs except IUs used to perform I/O operations in transport mode

3.2.18 disconnection

removal of a connection by closing one or both exchanges of an exchange pair or by closing a transport exchange

3.2.19 exchange pair

two FC-FS-4 exchanges between a channel and a control unit with sequence initiative in opposite directions that are linked together by the SB-6 ULP

3.2.20 extended CRC offset block

list of 1-word values identifying the CRC offsets and block lengths of segments of write transport data

3.2.21 Fabric

entity that interconnects Nx_Ports attached to it and is capable of routing frames by using the D_ID information in a Frame_Header (see FC-FS-4)

3.2.22 image

group of related processes behind a single N_Port (e.g. a single system or a single logical partition of a system)

3.2.23

inbound exchange

exchange of an exchange pair which originates from a control unit and that carries information to the channel

3.2.24 initiation IU

first IU of an exchange

3.2.25 I/O operation

execution of an operation specified by a CCW or a TCW

3.2.26 link-level functions

FC-FS-4 link-control mechanisms and basic-link services (BLS), FC-LS-3 extended-link services (ELS), and SB-6 link-control functions used to make operational and manage the physical and logical link between an entity acting as a channel and an entity acting as a control unit

3.2.27 mode of operation

identifies the protocols and functions of either command mode or transport mode, used by a channel and control unit to perform device-level functions.

3.2.28 N_Port

see FC-FS-4

3.2.29 N_Port_ID

see FC-FS-4

3.2.30 outbound exchange

exchange of an exchange pair which originates from a channel and that carries information from the channel to the control unit

3.2.31 persistent IU pacing

method for allowing an FC-SB-6 channel to retain a pacing credit for use at the start of execution of a channel program

3.2.32 SB-6 offline condition

condition recognized when a receiver transitions from the FC-FS-4 active state to the FC-FS-4 OLS receive state

3.2.33 transport command control block (TCCB)

variable length data structure that contains device-command words (DCWs) and associated control information that is sent from a channel to a control unit

3.2.34 transport command word (TCW)

a) control block which contains a request to perform an I/O operation in transport mode; or b) structure of a specific system architecture which specifies a TCCB to be sent from the channel to the CU

3.2.35 TCW channel program

single transport command word

3.2.36 TCW I/O operation

decoding, accepting, and executing a TCCB by an I/O device

3.2.37 transport exchange

single bi-directional exchange that carries the communication required between a channel and control unit to perform a device-level function

3.2.38 transport mode

mode of operation that is used to perform SB-6 device-level functions using transport exchanges

3.2.39 transport status block (TSB)

variable length data structure that contains status for a TCW I/O operation

3.2.40 ULP process

function executing within a channel or control unit which conforms to the Upper Level Protocol (ULP) immediately above the SB-6 service interface

3.3 Editorial Conventions

3.3.1 English Usage Conventions

In this standard, a number of conditions, mechanisms, sequences, parameters, events, states, or similar terms that do not have their normal English meaning are printed with the following conventions:

a) The first letter of each word in uppercase and the rest lowercase (e.g., Exchange, Class, etc.);

b) A term consisting of multiple words, with the first letter of each word in uppercase and the rest lowercase, and each word separated from the other by an underscore (_) character. A word may consist of an acronym or abbreviation which would be printed in uppercase. (e.g., N_Port); and

c) a term consisting of multiple words with all letters lowercase and each word separated from the other by a dash (-) character. A word may also consist of an acronym or abbreviation which would be printed in uppercase. (e.g., device-level, CUE-with-busy).

Certain words and terms used in this standard have a specific meaning beyond the normal English meaning.These words and terms are defined either in the definitions or in the text where they first appear.

Lists sequenced by lowercase or uppercase letters show no ordering relationship between the listed items. Lists sequenced by numbers show an ordering relationship between the listed items.

If a field or a control bit in a frame is specified as not meaningful, the entity which receives the frame shall not check that field or control bit.

3.3.2 FC Link Functions

In order to support the SB-6 protocol, the FC-FS-4 and FC-LS-3 functions referenced in this document are required to be supported by a channel, control unit and Fabric (if present) as specified in this document.

3.3.3 Bit Numbering

All words defined by FC-FS-4 and FC-LS-3 use the convention that bit 31 is the Most Significant Bit (MSB) and bit 0 is the Least Significant Bit (LSB). All words defined by the FC-SB-6 protocol use the convention that bit 0 is the MSB and bit 31 is the LSB. All bytes defined by FC-FS-4 and FC-LS-3 use the convention that bit 7 is the Most Significant Bit (MSB) and bit 0 is the Least Significant Bit (LSB). All bytes defined by the FC-SB-6 protocol use the convention that bit 0 is the MSB and bit 7 is the LSB. For FC-FS-4, FC-LS-3 and FC-SB-6, byte 0 is the most significant byte of a word, and byte 3 is the least significant. In all figures, tables and text of this document, the most significant bit of a binary value is shown as the left-most bit. Figure 1 gives a comparison between the FC-FS-4/FC-LS-3 convention and the FC-SB-6 convention.

Figure 1 – FC-FS-4/FC-LS-3 and FC-SB-6 Bit Numbering Conventions

3.3.4 Binary Notation

Binary notation may be used to represent some fields. Single bit fields are represented using the binary values zero and one. For multiple bit fields, the binary value is enclosed in single quotation marks followed by the letter b. For example, a 4-byte field containing a binary value may be represented as ‘00000000 11111111 10011000 11111010’b.

3.3.5 Hexadecimal Notation

Hexadecimal notation may be used to represent some fields. When this is done, the value is enclosed in single quotation marks and preceded by the word hex. For example, a 4-byte field containing a binary value of ‘00000000 11111111 10011000 11111010’b is shown in hexadecimal format as hex’00 FF 98 FA’.

3.4 Abbreviations, Acronyms, and Symbols

Abbreviations and acronyms applicable to this standard are listed below. See FC-FS-4 and FC-LS-3 acronyms associated with those specifications. For all other acronyms, refer to the index.

ABTS	Abort Sequence BLS
AS	address specific
BLS	FC-FS-4 Basic Link Service
CC	chain command
CCW	channel-command word
CD	chain data
CH	chaining
CHPID	channel path identifier
CI	channel initiated
CNP	CRC not provided
COB	CRC Offset Block
COC	continue on command immediate
CR	command retry
CRC	cyclic redundancy check
CRR	command response request
CTC	channel-to-channel
CTCA	channel-to-channel adapter
CU	control unit
DACK	device-level acknowledgment
DCW	device command word
DIB	device information block
DU	data chaining update
E	end

EE	early end
ELP	Establish Logical Path link-control IU
ELS	FC-LS-3 Extended Link Service
ES	extended status
FC	fibre channel
FC-FS-4	FC Framing and Signalling Interface-4 Specification
FC-LS-3	FC Link Services-3 Specification
FC-PI-6	FC Physical Interface Specification-6
FC-SB-6	Single Byte Command Code Sets Protocol-6 Specification
FLOGI	F_Port Login ELS
FQXID	Fully Qualified Exchange ID
IU	information unit (see FC-FS-4)
IUI	information unit identifier
LACK	Link Level Acknowledgment link-control IU
LBY	link level busy link-control IU
LESB	link error status block
LIRR	Link-Incident Record Registration ELS
LOGO	N_Port logout ELS
LPE	Logical Path Established link-control IU
LPR	Logical Path Removed link-control IU
LRC	longitudinal redundancy check
LRI	long record/immediate
LRJ	Link level Reject link control IU
LS_RJT	Link-Service Reject ELS Reply
PCI	program control interruption
PLOGI	N_Port Login ELS
PRLI	Process Login ELS
PRLO	Process Logout ELS
QSA	Query Security Attributes ELS
REC	Read Exchange Concise ELS
REX	repeat execute
RFCN	Registered Fabric Change Notification ELS
RLIR	Registered Link Incident Record ELS
RLP	Remove Logical Path link-control IU
RNID	Request Node Identification Data ELS
RRQ	Reinstate Recovery Qualifier ELS
RSCN	Registered State Change Notification ELS
RV	residual count valid
SCR	State Change Registration ELS
SDC	self-describing component
SLI	suppress length indication
SSS	synchronize send status
SYR	synchronize response
TCA	transport command area
TCAH	transport command area header
TCAX	transport command area extension
TCCB	transport command control block
TCH	transport command header
TCOB	Transfer CRC Offset Block command
TCW	transport command word
TIN	Test Initialization link-control IU
TINC	test initialization capability
TINCR	test initialization capability result

TIR	Test Initialization Result link-control IU
TSB	transport status block
TTE	Transfer TCA Extension command
ULP	upper level protocol

3.5 Keywords

3.5.1 mandatory: A keyword that indicates items required to be implemented as defined by this standard.

3.5.2 may: A keyword that indicates flexibility of choice with no implied preference.

3.5.3 model-dependent: A keyword describing an item (e.g. bit, field, code value, etc.) or a behavior (e.g., number of retries) that is not defined by this standard and may be vendor specific.

3.5.4 optional: A keyword that describes features that are not required to be implemented by this standard. However, if any optional feature defined by this standards is implemented, it shall be implemented as defined in this standard.

3.5.5 reserved: A keyword referring to bits, bytes, words, fields and code values that are set aside for future standardization. Their use and interpretation may be specified by future extensions to this or other standards. A reserved bit, byte, word or field shall be set to zero by the originator and ignored by the recipient.

3.5.6 shall: A keyword indicating a mandatory requirement. Designers are required to implement all such requirements to ensure interoperability with other products that conform to this standard.

3.5.7 should: A keyword indicating flexibility of choice with a preferred alternative; equivalent to the phrase “it is recommended”.

3.5.8 vendor specific: A keyword describing items (e.g., a bit, field, code value, etc.) that are not defined by this standard.

4 Structure and Concepts

4.1 Introduction

This clause provides an overview of the structure, concepts, and mechanisms used in Fibre Channel (FC), FC-FS-4, FC-LS-3 and FC-SB-6.

Communication over Fibre Channel takes place between a pair of N_Ports. FC-FS-4 and FC-LS-3 define all of the functions and protocols required to transfer information from one N_Port to another. Depending upon the configuration, the communicating N_Ports are between a channel and control unit or in the case of a channel-to-channel connection, between a channel and a channel emulating a control unit.

FC-SB-6 is a mapping protocol, referred to as an FC-4 that maps a particular upper level protocol (ULP) instance to FC-FS-4. The FC-SB-6 ULP is based on the Single-Byte Command Code Set (SBCON) and on FCP-4. See Annex E for additional information.

An N_Port supports an FC-4 mapping protocol and its associated ULP. More than one FC-4 and ULP instances of the same type or different type may be supported by the same N_Port.

4.2 FC-4 General Description

The FC-4 mapping layer uses the FC-FS-4 protocol to transfer Upper Level Protocol (ULP) information. Each FC-4 describes, through mapping rules, how ULP processes of the same ULP type interoperate. An example of an active ULP process is an FC-SB-6 I/O operation in progress at a device of a particular type such as a disk drive.

The protocols are described in terms of the elements visible in the stream of frames, sequences, and exchanges generated by a pair of communicating nodes transferring ULP information.

4.3 FC-SB-6 General Description

4.3.1 FC-SB-6 Instance

An FC-SB-6 ULP instance may be either an N_Port-based FC-SB-6 channel image, an N_Port-based SB-6 control-unit image, or an N_Port-based emulated control-unit image. Information associated with the execution of an I/O operation and the operation of a device is transferred between a channel image and control-unit image.

4.3.2 FC-SB-6 Protocols

Two levels of protocol are defined for this standard:

a) link-level; and

b) device-level.

FC-SB-6 protocols are classified as either link-level or device-level, depending on whether they are for the purpose of managing a logical path and exchanging information over that path or for the purpose of managing the execution of an I/O operation.

The execution of an I/O operation requires both FC-SB-6 link-level and device-level protocols at the channel and control unit. Information exchanged between a channel and control unit as a result of executing an I/O operation is transferred under the control of device-level protocols, which rely on link-level protocols for sending and receiving frames (see 4.7.2 and 4.7.3).

4.4 Channel-Path Elements

4.4.1 Overview of Channel-Path Elements

The channel path provides the communication path between two systems. The physical elements that make up a channel path are a channel, possibly a Fabric, a control unit, and one or more links.

4.4.2 Channel

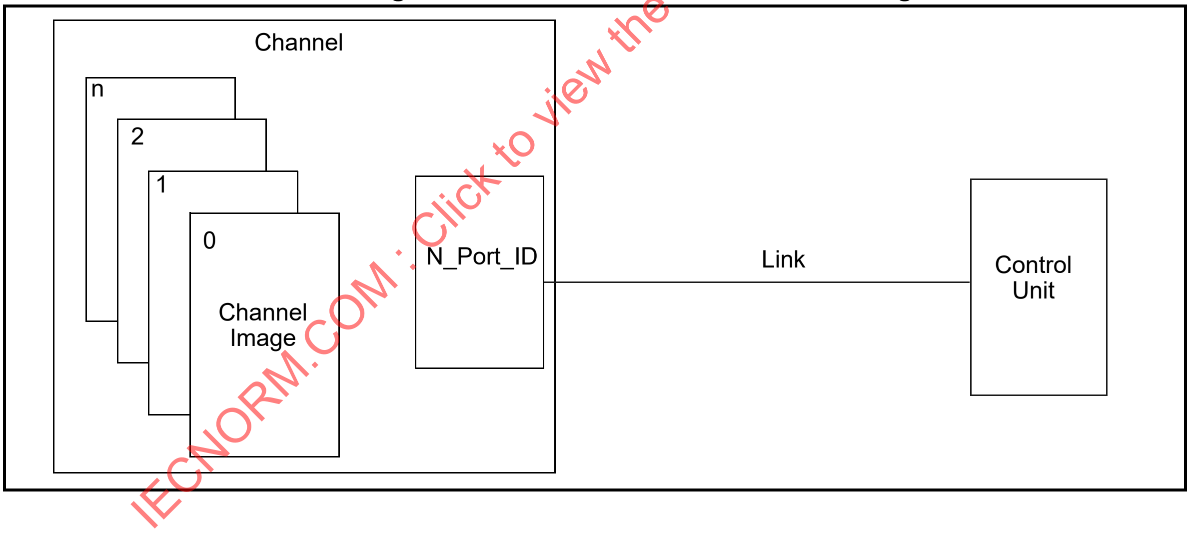
A channel provides the functions specified by FC-FS-4, FC-LS-3, and FC-SB-6, and performs their prescribed protocols. Each channel shall provide one N_Port. When the N_Port of a channel is shared by multiple processes, such as multiple operating systems, each channel path shall be logically and separately represented within the channel. This logical representation is termed a “channel image” (see 4.4.3). A channel may also be capable of providing control unit emulation (see 4.5.6.2).

4.4.3 Channel Image

A channel image has the logical appearance of a channel. Each channel image appears to be an independent channel although all channel images on a specific channel share the same N_Port_ID and physical paths. The N_Port of a channel performs certain functions for all sharing channel images (e.g., link synchronization and acquisition of address identifiers) and may perform functions for a single sharing channel image (e.g., sending frames during the performance of link-level and device-level functions). The N_Port of a channel common to multiple channel images may perform functions for multiple channel images simultaneously by multiplexing work on each function and interleaving frames for each function on the link as allowed by FC-FS-4.

Figure 2 shows multiple channel images which are sharing the same N_Port_ID.

Figure 2 – Channel Path with Channel Images



Either single or multiple channel images may exist at an N_Port. Each channel image denotes one FC-SB-6 channel ULP instance.

NOTE 1 – Channel images represent different logical channel paths even though the channel images share the same N_Port. Therefore, there is one channel per channel path.

4.4.4 Control Unit

A control unit provides the logical capability necessary to operate and control one or more devices and adapts, through the use of common facilities, the characteristics of each device to an N_Port provided by the channel. These common facilities which include N_Port functions and may include FC-SB-6 functions provide for the execution of I/O operations, indications concerning the status of the device and control unit, control of the timing of data transfer over the channel path, and certain levels of device control.

A control unit may have more than one N_Port in order to allow attachment to more than one link, each from a different channel or from a different F_Port on the same or a different Fabric. When a control-unit N_Port is attached to a link from a Fabric, the control unit and its devices may be physically accessible over that N_Port to all channels also attached to links from that Fabric.

When an N_Port on a control unit is shared among multiple channel paths, each channel path shall be logically represented separately within the control unit. This logical relationship is called the logical path (see 4.5.5). A control unit shall provide at least one logical path for each operational N_Port_ID, however, the number of logical paths that a control unit permits is model dependent.

A control unit may also be capable of supporting N_Ports that act as channel paths.

NOTE 2 – Certain ULPs have a dependency upon the existence of a single path to a control unit for a given channel for purposes of managing the I/O configuration. In order to satisfy this dependency, it is recommended that a control unit have only one N_Port which is configured to communicate with a given channel.

NOTE 3 – A control unit may have more than one N_Port, each configured to communicate with a different channel.

NOTE 4 – A control unit may be connected to more than one channel at the same time. When a control unit is connected to more than one channel at the same time, each connection may be for the same or a different device.

4.4.5 Control-Unit Image

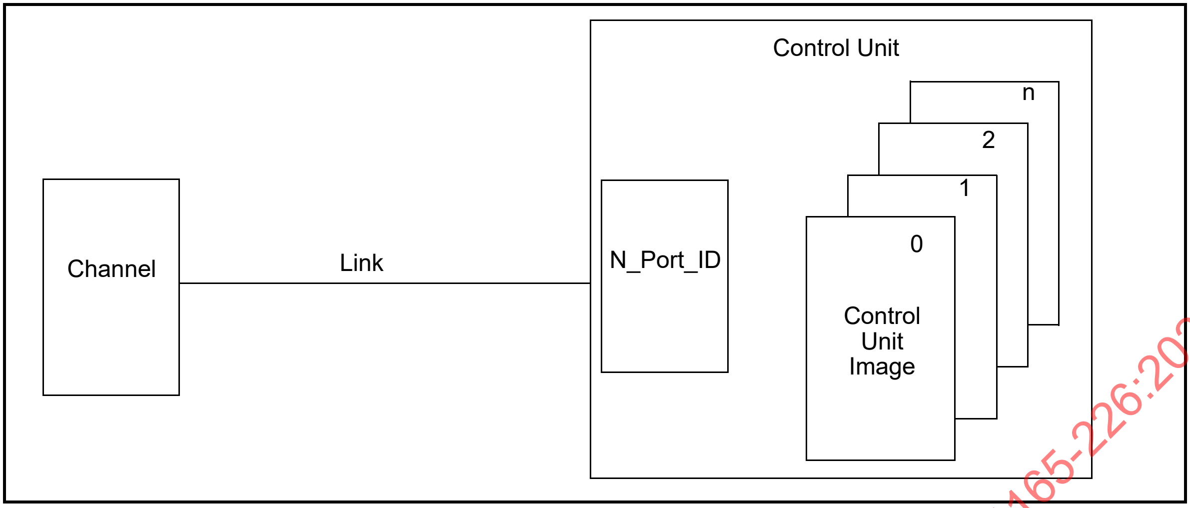
A control-unit image has the logical appearance of a control unit. Each control-unit image appears to be an independent control unit, although all control-unit images common to one control unit may share the same common facilities and N_Ports. These common facilities may provide some or all of the FC-SB-6 functions and protocols. Those FC-SB-6 functions and protocols not provided by the common facilities shall be provided by the individual images.

Up to 256 devices may be attached to each control-unit image.

FC-SB-6 protocols operate for each control-unit image independent of all other control-unit images except for resolving contention for the common facilities shared among control-unit images. The N_Port_ID of a control-unit common to multiple control-unit images may perform functions for multiple control-unit images simultaneously by multiplexing work on each function and interleaving frames for each function on the link as allowed by FC-FS-4.

Figure 3 shows multiple control-unit images which are common to a single N_Port_ID.

Figure 3 – Channel Path with Control-Unit Images



NOTE 5 – Each control-unit image is architecturally identical, even though each image may provide a different device type. The presence of multiple control-unit images becomes apparent only through the logical addresses used when logical paths are established (see 4.7.4 and 4.5.5).

4.4.6 Link

The link is as described by FC-PI-2, FC-PI-5, FC-PI-6, FC-PI-6P, FC-PI-7 and FC-FS-4.

4.5 Channel-Path Configurations

4.5.1 Channel-Path Configuration Overview

Support for point-to-point configuration and Fabric configuration is provided.

4.5.2 Point-to-Point Configuration

A channel path that consists of a single link interconnecting one or more control-unit images to one or more channel images forms a point-to-point configuration. A point-to-point configuration shall be permitted between a channel and control unit only when a single control unit is defined on the channel path or when multiple control-unit images all share the same N_Port. Figure 4 shows a point-to-point configuration consisting of a single channel image and single control-unit image, and figure 5 shows a point-to-point configuration consisting of multiple channel images and control-unit images.

Figure 4 – Point-to-Point Channel-Path Configuration (Single Logical Image)

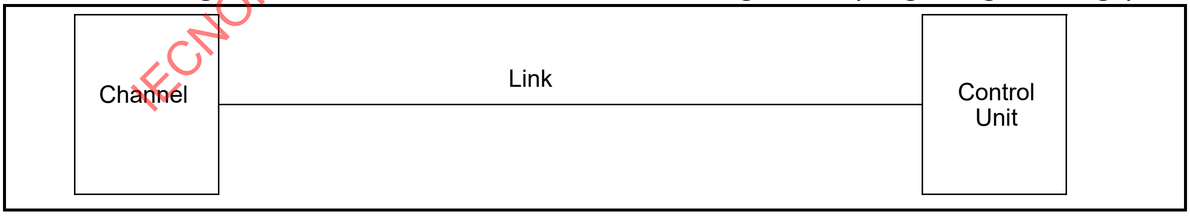
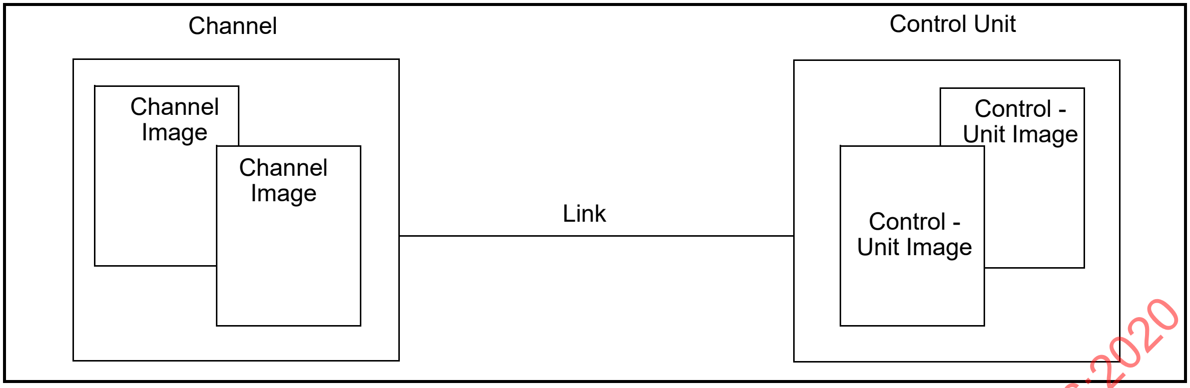


Figure 5 – Point-to-Point Channel-Path Configuration (Multiple Logical Images)



The channel N_Port and the control-unit N_Port resolve contention among the logical images for access to the link.

A maximum of one link shall be attached to the channel in a point-to point channel-path configuration. The maximum number of control-unit images that shall be addressed over the link is 256; therefore, the maximum number of devices that shall be addressed over point-to-point channel path configuration is equal to 256 control-unit images times 256 devices per control-unit image, or 65536.

4.5.3 Fabric Configuration

A channel path that consists of one link which interconnects one or more channel images with a Fabric and one or more links, each of which interconnects the Fabric with one or more control-unit images, forms a Fabric configuration.

Multiple channel images and multiple control-unit images may share the resources of the FC links and the Fabric, such that multiplexed I/O operations may be performed.

There are two types of Fabric configurations: single-switch and multi-switch. In a single-switch Fabric configuration, all channels and control units are attached to a single switch (see figure 6). In a multi-switch Fabric configuration, channels and control units are attached to multiple, interconnected switches (see figure 7).

Figure 6 – Single-switch Fabric Channel-Path Configuration (Multiple Channel Images)

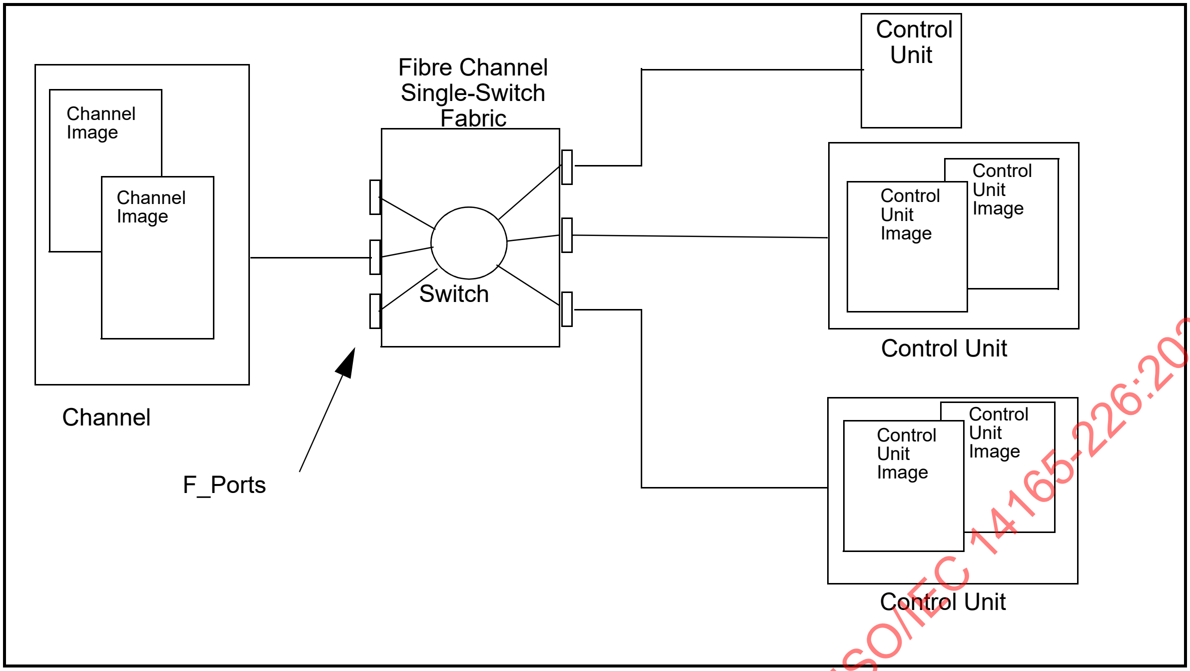
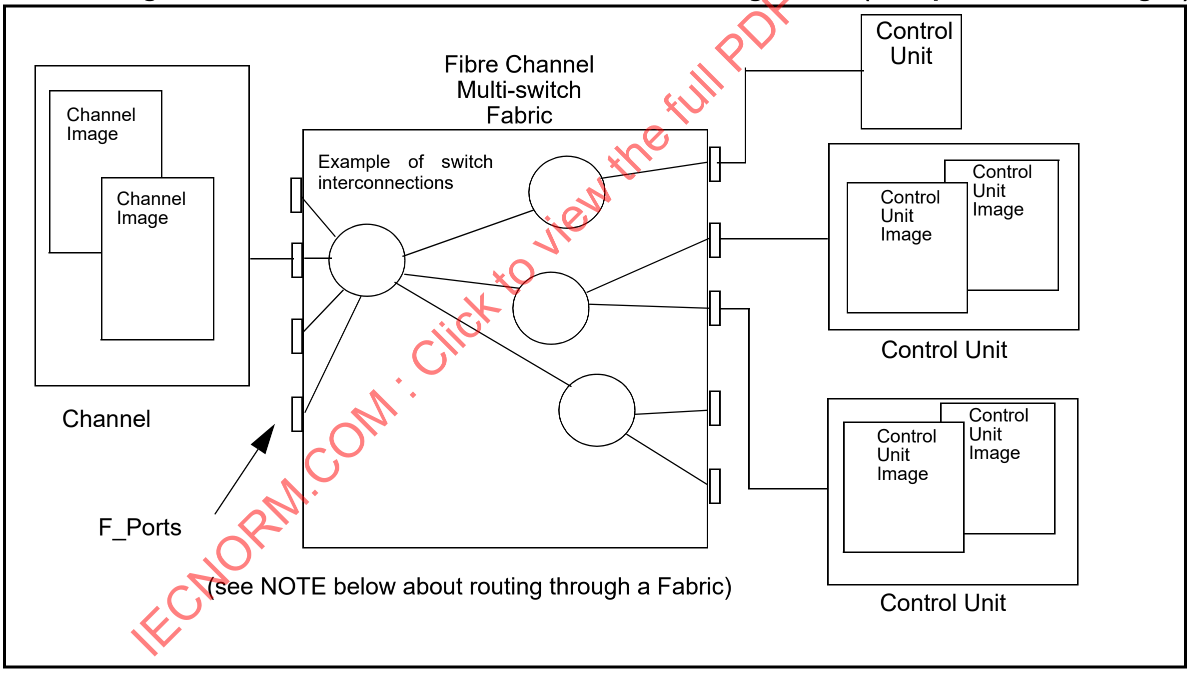


Figure 7 – Multi-switch Fabric Channel-Path Configuration (Multiple Channel Images)



Channels and control units may be attached to the links from the Fabric in any combination, depending on configuration requirements and depending on available resources in the Fabric. Sharing a control unit through a Fabric means that communication with the control unit may take place over one N_Port in the case where the control unit has only one link to the Fabric or over multiple N_Ports in the case where a control unit has more than one link to the Fabric.

A maximum of one link shall be attached to the channel in a Fabric configuration. The same control unit and device addressing capability exists as for the point-to-point configuration, however the attachment flexibility is greatly increased with a Fabric through the capability to use the 24 bit N_Port address, to access multiple control units.

For a detailed description of the Fabric refer to FC-SW-6. The N_Port of the channel or control unit attaches to an F_Port of the Fabric. The F_Port is required to support at least Class 2 and Class 3 service.

NOTE 6 – This standard does not limit the number of switches in a Fabric but some implementations may limit the number or switches in the route between an N_Port of any channel and an N_Port of any control unit due to configuration management considerations.

4.5.4 Physical Path

The communication path between a channel and a control unit is composed of two different parts, the physical channel path (physical path) and the logical channel path (logical path). The physical path is the link, or the interconnection of two or more links by a Fabric, that provides the physical transmission path between a channel and a control unit. The logical path is the relationship that exists between a channel and a control unit for device-level communication during execution of an I/O operation and presentation of status.

4.5.5 Logical Path

A logical path is the relationship established between a channel image and control-unit image which identifies a communication path over which device-level information may be transferred and to which certain device-level allegiances may be associated. The logical path is established as part of the channel and control unit initialization procedures by the exchange of FC-SB-6 link-control IUs.

When the logical path is established, device-level communication is allowed on that logical path. All device-level protocols depend upon the existence and identity of logical paths. Device-level protocols are executed over an established logical path by means of the exchange of information units between the channel and the control unit. When a logical path is not established, only link-level communication is permitted.

A logical path is identified within a channel or control unit by the combination of a 24-bit N_Port address identifier assigned to the channel, a 24-bit N_Port address identifier assigned to the control unit, an 8-bit logical address assigned to the channel image establishing the logical path, and an 8-bit logical address assigned to the control-unit image to which the logical path is being established. Both the channel and the control unit recognize a logical path by the same combination of addresses. The two 24-bit address identifiers define the N_Ports to which the images are associated, and the two 8-bit logical addresses identify the images for which the logical path exists.

A channel image may have one or more logical paths to each control-unit image. The number of logical paths a channel or control unit permits depends on system requirements and model capabilities.

Even though multiple logical paths may be associated with the same N_Port on the control unit the channel paths represented by these logical paths are treated as if each were associated with a separate N_Port interface; each logical path represents a logical relationship to a channel.

Allegiances, I/O operations, system resets, and path groups for a particular system are identified by means of the logical path established by the channel for that system. To a control unit, each logical path represents a different channel path or logical subdivision of a channel path.

When multiple control-unit images are provided, their presence is apparent only through associated logical paths.

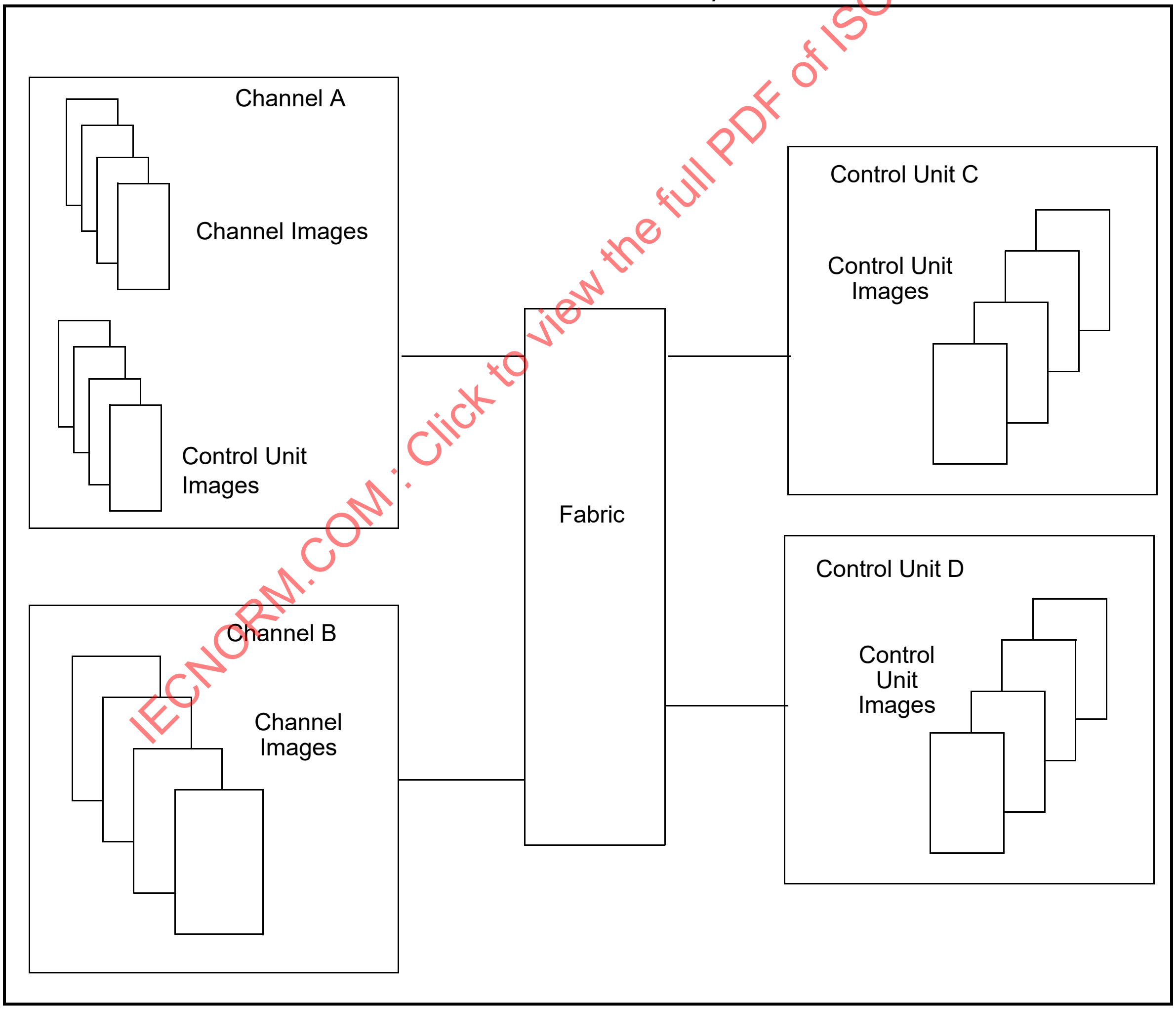
4.5.6 Channel-to-Channel Communication

4.5.6.1 Channel-to-Channel Communication Overview

A channel supporting channel-to-channel communication shall be able to accept a request to establish one or more logical paths and shall be able to provide all of the functions of a control-unit image on logical paths on which communication is to take place. In addition to providing all of the functions required of a channel for other logical paths, a channel supporting channel-to-channel communication shall provide all of the functions required of a control unit on a logical path for which it has accepted an Establish Logical Path (ELP) IU (see 6.4.2). When a channel supports channel-to-channel communication, the N_Port of the channel is also the N_Port of the emulated control unit.

Figure 8 shows a Fabric configuration in which a control-unit image of one channel, referred to as "channel A" is communicating with another channel, referred to as "channel B." In the configuration shown, one or more of the control-unit images of channel A has accepted a request from channel B to establish a logical path, and the control-unit image within channel A is providing all of the functions required of a control unit on the logical path to a corresponding channel image of channel B. Both channel A and channel B may also have established logical paths with other control units attached to the Fabric.

Figure 8 – Fabric Channel-Path Configuration (Multiple Channel Images, Channel-to-Channel Connection)



4.5.6.2 FC-SB-6 Channel-to-Channel Adapter

A FC-SB-6 channel-to-channel adapter is a channel capable of emulating a control unit and is used by a program in one system to communicate with a program of another system. The FC-SB-6 channel-to-channel adapter provides the data path and synchronization for a data transfer between two channels; however, in so doing, it presents a view as though it were a control unit including support of requests for logical paths and all of the functions of a control-unit image (or images) on logical paths over which communication is to occur.

4.6 Information Transfer

FC-SB-6 information is transferred on an FC link in FC-4 information units. An information unit is a collection of data that is organized according to a particular structure depending on the function being performed or the data content. The types and structures of the IUs defined for this standard are described in 8.3. An information unit is transferred in a sequence which shall be sent as one or more FC-FS-4 device-data frames.

4.7 Protocols

4.7.1 Protocol Overview

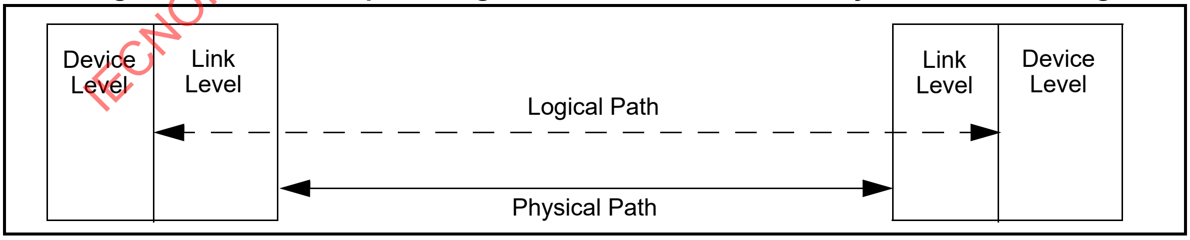
The link-level protocols describe the physical characteristics of a channel path and the associated protocols required for the transmission and reception of frames. The link-level protocols for this standard are defined in FC-FS-4, FC-LS-3, and in clause 6. The device-level protocols are associated with the execution of an I/O operation for a specific device. The device-level protocols for FC-SB-6 are defined in this standard and provide multiplexing at the frame level, streaming of commands and data, and the transport of multiple I/O commands in a single IU. The frame level multiplexing provided is based on Fibre Channel protocols. This multiplexing allows for multiple simultaneous exchanges, each concurrently transferring information, between a channel and control unit over the same N_Port to different devices.

The presence of multiple channel images or multiple control-unit images only becomes apparent through the logical paths that are established. All device-level protocols depend on the existence of and identity of these logical paths. When the logical path is known, the particular logical images for which the link-level and device-level facilities are dedicated at any one time is also known.

The execution of an I/O operation requires link-level and device-level protocols to be present in both the channel and the control unit. Information exchanged between a channel and control unit as a result of executing an I/O operation is transferred under the control of the device-level protocols with the aid of the link-level protocols for sending and receiving frames. The failure to satisfy protocols at either level results in an error being recognized.

Figure 9 shows the relationship between the link level, device level, physical path, and logical path.

Figure 9 – Relationships among Link Level, Device Level, Physical Path, and Logical Path



4.7.2 Link Level Protocol

Link-level protocols refer to FC-FS-4 Basic Link Services (BLS), FC-LS-3 Extended Link Services (ELS) and FC-SB-6 link-level functions.

Link-level protocols are used to establish and maintain the physical and logical paths in order to provide for transmission and reception of frames and primitive sequences. They include functions such as acquiring address identifiers, establishing frame credit, generating the basic frame structure, coordinating the protocols for frame transmission, and checking the integrity of information sent in frames (see FC-FS-4, FC-LS-3 and clause 6).

Certain functions require the exchange of information between the channel and control unit at the link level. These functions are performed by means of link-level protocols (e.g., initialization). Before communication over a channel path is allowed to occur, and before an I/O operation may be executed, some form of initialization shall be completed according to the requirements of the system and according to the specifications of the architectural definition (see FC-LS-3 for link initialization and login definitions).

Once the N_Port of a channel has performed initialization procedures for a control unit, including logging in and establishing a logical path to a control-unit image, that control-unit image is considered operational and capable of executing I/O operations over that channel path.

4.7.3 Device Level Protocols

Device-level protocols allow for transfer of information specifically related to an I/O operation, transfer of status of a device or control unit, and recovery when errors are detected by the channel or the control unit.

The types of device-level information transferred between a channel and control unit that use device-level protocols are: commands, data, control information, and status. Commands are provided by the channel program being executed. Data is the information associated with the command being executed that is to be transferred to or from the device. Control information includes functions that manage the transfer of data between the channel and control unit and functions that manage execution of the I/O operation. Status describes the results of the completion of an I/O operation (successful or unsuccessful) or provides information not associated with an I/O operation that is to be reported to the channel.

Device-level information is transferred between a channel and a control unit in FC-SB-6 information units (see 8.3). FC-SB-6 information units are transferred using link-level functions, device-level functions, link-level protocols and device-level protocols (e.g.,when the channel receives initiative to start an I/O operation in command mode, the device-level functions and device-level protocols obtain the command and other parameters from the current CCW and insert them into the appropriate fields within a command IU). When the command IU is ready for transmission, link-level functions and link-level protocols provide additional information (e.g., address identifiers and exchange ID in the frame header) and then coordinate the actual transmission of the frame on the channel path.

4.7.4 Addressing

Two levels of addressing are used for the link-level and device-level structure. All control units and channels use both levels of addressing. Link-level addressing identifies the N_Port, which identifies the physical path within the channel-path configuration to be used for communication between a channel and a control unit. Link-level addressing also identifies the channel images or control-unit images associated with an N_Port (see FC-FS-4).

Device-level addressing identifies the device to the channel or control unit, once the physical path, as designated by the link-level addressing, has been determined.

Link-level addressing is considered to be the first level of addressing because it determines the N_Port, physical path, and the sharing logical image; device-level addressing is considered to be the second level of addressing because it determines the device once the N_Port and logical image are determined. Both levels of addressing have specific address assignment requirements.

Link-level addressing requires each N_Port to be assigned a 24-bit address, called the address identifier. The address identifier of the sender of a frame is in the S_ID field in the FC-FS-4 header, and the address identifier or the recipient of a frame is in the D_ID field of the FC-FS-4 header (see 5.5.3). An N_Port that does not have an address identifier assigned is *unidentified*; an N_Port that has an address identifier assigned is *identified*. The assignment of an address identifier to an N_Port occurs when the N_Port performs initialization and the required login procedure (see FC-LS-3). When an unidentified N_Port performs initialization, it acquires its address identifier through the procedures defined for the login process.

Link-level addressing also requires that each channel and control-unit image be assigned an 8-bit address, called the logical address. The logical addresses of the sender and receiver of an IU are in the FC-SB-6 header field of the IU (see 8.4). A channel image or control-unit image is assigned a logical address when it is created. A logical address shall be unique at a channel or control unit N_Port_ID but need not be unique on a channel path. The assignment of logical addresses and the method by which this assignment is performed are model dependent.

When either multiple channel images or multiple control-unit images are created, they share a single N_Port_ID for each link, and therefore, the same respective address identifier. A maximum of 256 channel images and 256 control-unit images shall share the same N_Port_ID. The combination of the address identifier and the logical address determines the image to which the device-level addressing applies.

Device-level addressing depends on the assignment of an 8-bit address, called the device address, to every device. A device address shall be unique on a logical path on a control unit but need not be unique on a channel path because the resultant combination of the N_Port address identifier, control-unit logical address, and the device address uniquely identifies a device on a channel path. The assignment of device addresses and the method by which this assignment is performed are model dependent (see 8.4.5).

Control units that attach to more than one link provide an N_Port per link, with each N_Port having an assigned address identifier.

All IUs sent on a channel path rely on link-level addressing information to determine the correct physical path to the destination N_Port_ID and, for certain IU types, to determine the image associated with that N_Port_ID. Some IUs require only link-level addressing while others require both link-level addressing and device-level addressing. An IU sent to perform a link-level protocol requires only link-level addressing, and an IU sent to perform a device-level protocol requires link-level addressing and in some cases device-level addressing. Device-level addressing identifies the device that is the source or destination of the information in the IU. For device-level protocols that does not use device-level addressing, the protocol relies on the fully qualified exchange identifier (FQXID) associated with the exchange to identify the logical path and device. The FQXID is provided in the FC-FS-4 header and is composed of the source port identifier, the destination port identifier, the OX_ID field value, and the RX_ID field value (see FC-FS-4).

5 FC-FS-4 Link Control

5.1 FC-FS-4 Link Control Overview

FC-FS-4 provides services which support the transfer of data between N_Ports in frames which consist of a frame header and a payload. This clause describes how FC-FS-4 services are used to perform FC-SB-6 link-level functions and device-level functions, and defines how the fields in the frame header are used to perform these functions. For a general description mechanisms and frame header fields which are unrelated to this standard, see FC-FS-4.

5.2 Class of Service

Class 3 service shall be used when sending all FC-SB-6 IUs except for the test initialization IU (see 6.4.7). Class 2 service shall be used when sending the test initialization IU because the N_Port receiving the test initialization IU may be in a non-operational state. When this is the case, Class 2 provides the means for the Fabric to provide timely notification of this fact, thereby avoiding costly timeouts and retry operations.

The FC-SB-6 ULP requires the Fabric to indicate support for sequential delivery during Fabric login (see clause 7).

NOTE 7 – Since this standard requires the test initialization IU to be sent in Class 2, N_Ports which support this standard are required to indicate support for Class 2 at login. See FC-LS-3 for additional information.

5.3 Buffer-to-Buffer Credit Reclamation

During operation over extended periods of time, errors may occur which result in the loss of one or more R_RDY primitive signals. The loss of R_RDY primitive signals affects the available buffer-to-buffer credit at an N_Port and as a result, may affect that N_Port’s ability to send frames and maintain optimum performance. Designs should be tolerant of the loss of some R_RDY primitive signals for short periods of time but also shall be capable of taking some action to offset the cumulative effect of errors over longer periods of time. The cumulative loss of R_RDY primitive signals over long periods of time may result in a reduction in available buffer-to-buffer credit to a point where performance is significantly affected or, in the extreme case where the available buffer-to-buffer credit goes to zero, the inability to send a frame, and an FC-FS-4 link timeout error.

To prevent the loss of R_RDY primitive signals from resulting in a performance degradation or FC-FS-4 link timeout error, N_Ports shall perform a model-dependent buffer-to-buffer credit reclamation procedure which reinitializes the available buffer-to-buffer credit at both ends of the link. The preferred reclamation procedure is specified in FC-FS-4.

When the preferred buffer-to-buffer credit reclamation procedure is not supported, an alternative procedure shall be performed periodically at model-dependent time intervals. The time interval of the model-dependent procedure should be as large as possible provided that buffer-to-buffer credit reclamation is performed before the loss of R_RDY primitive signals results in a performance degradation. To minimize the number of IUs lost as a result of buffer-to-buffer credit reclamation, the buffer-to-buffer reclamation procedure is not initiated when open exchanges exist, and an effort is made to quiesce the link before the procedure is performed.

NOTE 8 – When the preferred buffer-to-buffer credit reclamation procedure is not performed, the channel may quiesce the link before performing buffer-to-buffer credit reclamation by not initiating new exchanges and allowing all channel programs to complete and all exchanges to end. Control units may quiesce the link by completing all channel programs which are in process and by returning control unit busy status to new commands or control functions other than a system reset or purge path. Since efforts made to quiesce the link do not guarantee that no IUs are lost, occasional errors resulting from buffer-to-buffer credit reclamation should be expected.

NOTE 9 – When the preferred buffer-to-buffer credit reclamation procedure is not performed, implementations should use the FC-FS-4 link reset primitive sequence protocol to provide buffer-to-buffer credit reclamation. In addition, it is preferred that data queued in receive and transmit buffers before the protocol is initiated as well as data received into buffers during the time when the protocol is being performed be preserved and processed normally when the protocol is completed. For additional information on the link reset primitive sequence protocol, see FC-FS-4.

NOTE 10 – When the preferred buffer-to-buffer credit reclamation procedure is not performed, the suggested default time interval between the performance of each buffer-to-buffer credit reclamation procedure is approximately 24 hours. A means should be provided to adjust the time interval and randomly vary the time interval by up to 10% so that reclamation does not occur at predictable time intervals such as the same time each day. For Fabric configurations, a means should be provided which minimizes the probability of simultaneous buffer-to-buffer credit reclamation by a large number of N_Ports because of the system performance impact which results. The means used to avoid simultaneous buffer-to-buffer credit reclamation is model-dependent but may include the use of a slightly different time interval by each N_Port or by causing the time interval of each N_Port to begin at a different time.

NOTE 11 – When the preferred buffer-to-buffer credit reclamation procedure is not performed, unnecessary execution of an alternative buffer-to-buffer credit reclamation procedure should be avoided. An implementation should defer buffer credit reclamation for a model-dependent time interval after performing a procedure such as login or link initialization which initializes the available buffer-to-buffer credit. In a point to point configuration, only one of the N_Ports is required to initiate the buffer-to-buffer credit reclamation procedure because the procedure initializes the available buffer-to-buffer credit at both ends of the link. F_Ports are not required to perform the procedure since performance of buffer reclamation by the attached N_Port reinitializes the available buffer-to-buffer credit at both the N_Port and the attached F_Port.

5.4 FC-SB-6 Sequences and Exchanges

5.4.1 FC-SB-6 Sequences

When receiving frames from the link, FC-FS-4 provides a mechanism which assembles sub-blocks of data contained in the payloads of one or more frames into a single data block called a sequence. Each FC-4 defines the contents of the sequences which are used for ULP functions. When the contents and usage of a sequence are defined by a ULP, it is referred to as an information unit (IU). For this standard, IUs contain commands, device-level controls, link controls, and related functions. The IUs defined by this standard are summarized in Table 1, and the contents of FC-SB-6 IUs conform to the general structures in 8.3. The protocols for using IUs for FC-SB-6 link-level functions and device-level functions are found in clause 6 and clause 9, respectively.

5.4.2 FC-SB-6 Exchanges

An exchange pair consisting of two unidirectional exchanges, one used by the channel to send IUs and one used by the control unit to send IUs, are required for all FC-SB-6 link-control functions and for all FC-SB-6 device-level functions that are executed in command mode. A single bi-directional exchange, referred to as a transport exchange, is required for all device-level functions executed in transport mode. Exchange pairs and transport exchanges are described in the following sections.

5.4.2.1 FC-SB-6 Exchange Pairs

IUs which a channel sends during the execution of an FC-SB-6 link-control function or the execution of an FC-SB-6 device-level function in command mode are restricted to one exchange, and IUs which a channel receives during the operation are restricted to a different exchange. The exchange on which the channel sends IUs is referred to as the outbound exchange, and the exchange on which the channel receives IUs is referred to as the inbound exchange.

Inbound exchanges and outbound exchanges shall transfer IUs in a single direction. When both an outbound exchange and an inbound exchange simultaneously exist between a channel and a control unit for the execution of the same link-level function or device-level function, an exchange pair is said to exist, and the control unit is said to be connected to the channel. A channel program which is executed in a single connection uses only one exchange pair. If the connection is removed by the closing of the exchanges during the channel program, a new exchange pair shall be required to complete the channel program.

5.4.2.1.1 General Rules for Initiating Exchange Pairs

A channel shall initiate an exchange pair by sending an initiation IU as an unsolicited command or unsolicited control information category. A control unit shall initiate an exchange pair by sending an initiation IU as an unsolicited control or unsolicited data information category (see Table 1).

5.4.2.1.2 Exchange Pairs for FC-SB-6 Link Control Functions

A control-unit image that initiates an exchange pair for an FC-SB-6 link-control function shall wait for a response to the initiation IU before it initiates another exchange pair with the N_Port of the same channel if the exchange pair to be initiated is:

a) for another FC-SB-6 link-control function, or

b) for a device-level function not associated with a specific device.

5.4.2.1.3 Exchange Pairs for Device-Level Functions

A control-unit image that initiates an exchange pair for a device-level function not associated with a specific device shall wait for a response to the initiation IU before it initiates another exchange pair if the exchange pair to be initiated is:

a) with the same channel image for a device-level function not associated with a specific device; or

b) with the N_Port of the same channel for an FC-SB-6 link-control function.

A channel or control unit may initiate multiple exchange pairs, each for a different device or for the same device provided that there is no more than one exchange pair or transport exchange for a specific device on a logical path. A new exchange pair for a specific device on a logical path shall not be initiated when any of the following conditions apply:

a) an exchange pair or transport exchange already exists for the same device on the same logical path; or

b) the channel or control unit initiating the new exchange pair has initiated another exchange pair for the same device on the same logical path and not received a response.

If either of the above conditions apply, the channel or control unit shall wait for the existing exchange pair or transport exchange to be closed before initiating the new exchange pair for the device on the logical path.

5.4.2.1.4 Exchange Pair Processing

In order to avoid errors caused by lost IUs or by the inability to send an IU, sufficient resources should be held in reserve in order to support new exchange pairs for unexpected events. If an IU initiating a new exchange pair is received, sufficient resources should be available to properly receive the IU and to initiate a new exchange in response. If an unexpected event requiring a new exchange pair to be initiated occurs, resources should be available for supporting the new exchange pair.

After a channel or control unit has closed one of the exchanges of an exchange pair for a device on a logical path, an abnormal condition may result in the recognition of a valid initiation IU for the same device on the same logical path before the other exchange of the previous exchange pair has been closed. When this occurs and the initiation IU is not a purge path IU, the IU may be accepted and held until both exchanges of the previous exchange pair have been closed or a link-busy condition may be recognized (see 6.4.10 and 9.5.1).

5.4.2.2 FC-SB-6 Transport Exchanges

IUs which a channel and control unit send during the execution of an FC-SB-6 device-level function that is performed in transport mode are restricted to a single, bi-directional exchange referred to as a transport exchange. Transport exchanges are not used to execute link-level functions.

5.4.2.2.1 General Rules for Initiating a Transport Exchange

A channel shall open a transport exchange by sending a transport-command IU as an Initiation IU. A control unit shall not open a transport exchange (see table 1).

5.4.2.2.2 Transport Exchanges for Device-Level Functions

A channel may open multiple transport exchanges, each for a different device or for the same device on different logical paths. A new transport exchange may be opened for a specific device on a logical path when a transport exchange already exists for that device and logical path only to perform an interrogate operation (see 9.3.3.2); otherwise, the channel shall not open a new transport exchange to a device and logical path when a transport exchange or exchange pair is open for that device and logical path. When a transport exchange or exchange pair is open for a device and logical path, the channel shall wait for the existing transport exchange or exchange pair to be closed before initiating the new transport exchange for the device on the logical path.

5.5 FC Frame Header Fields

5.5.1 Frame Header Field Overview

The FC frame header identifies the source and destination of a frame, the exchange and sequence to which a frame belongs, the order in which a frame was sent, and the type of information in the frame payload. The frame header consists of a frame routing field (R_CTL), source link address field and destination link address field (S_ID and D_ID), an FC-4 identification field (TYPE), an optional header control field (DF_CTL), fields which identify and control FC-FS-4 sequences and exchanges (F_CTL, OX_ID, RX_ID, SEQ_ID, SEQ_CNT), and a parameter field whose function depends on the frame type. The usage of these fields for frames which are part of an IU is described in5.5. The frame header format is described in FC-FS-4. For a definition of the settings of frame header fields in frames which contain FC link-control functions and BLSs, see FC-FS-4. For a definition of FC ELSs, see FC-LS-3.

5.5.2 R_CTL Field

The R_CTL field contains two sub-fields, the routing bits and the information field. The routing bits identify the frame payload as FC-4, FC-FS-4 or FC-LS-3. For all frames used to send an IU, the routing bits shall be set to ‘0000’b to indicate that the payload contains FC-4 Device_Data.

When the routing bits are set to ‘0000’b and the frame is used to send an IU, the information field contains an information category field which identifies the category of the information in the payload (see FC-FS-4 for a list of information categories and their corresponding values of the information field). All frames in an FC-SB-6 IU shall contain the same information category. The information category of an IU depends on the contents of the IU and the conditions under which the IU is sent. Table 1 summarizes the information categories which shall be used to send each IU.

Table 1 – Information Categories of FC-SB-6 IUs

FC-SB-6 IU Name	Information Category	Primary Content
Command IU	Unsolicited Command	Command DIB or Command/Data DIB
Solicited Data IU	Solicited Data	Data DIB or Status DIB
Unsolicited Data IU	Unsolicited Data	Status DIB
Solicited Control IU	Solicited Control	Link control DIB or Control DIB
Unsolicited Control IU	Unsolicited Control	Link Control DIB or Control DIB
Transport Command IU	Unsolicited Command	Transport Command Control Block (TCCB)
Transport Data IU	Solicited Data	Data
Transport Response IU	Command Status	Status and Extended Status
Transfer Ready IU	Data Descriptor	Data Offset and Length
Transport Confirm IU	Solicited Control	none

5.5.3 D_ID and S_ID Fields

Each channel and control unit is assigned a unique N_Port ID during the initialization procedure (see clause 7). When sending an IU, the D_ID field shall be set to the N_Port ID of the recipient of the frame, and the S_ID field shall be set to the N_Port ID of the sender of the frame.

5.5.4 CS_CTL

When the F_CTL bit 17 is set to zero, Word 1 bits 24-31 of the Frame_Header is defined as the CS_CTL field. The CS_CTL field is defined by FC-FS-4.

When the F_CTL bit 17 is set to one, Word 1 bits 24-31 of the Frame_Header is defined as the Priority field. The Priority field is defined by FC-FS-4. Support for the use of non-zero Priority values is indicated in F_Port_Login (FLOGI) and N_Port_Login (PLOGI) (see FC-LS-3), and when it is indicated by the setting of F_CTL bit 17, FC-SB-6 has additional requirements on the setting of the Priority field for command mode operations (see 9.2.5) and transport mode operations (see 9.3.6).

5.5.5 TYPE Field

The TYPE field identifies the FC-4 protocol which defines the frame payload. This field shall be set to hex’1B’ in all frames of a command-mode IU that are sent by a channel. This field shall be set to hex’1C’ in all frames of a command-mode IU that are sent to a channel. All frames of a transport-mode IU sent by the channel or to the channel shall have the TYPE field set to hex‘08’.

If a channel which does not provide control-unit emulation receives an ELP IU with the TYPE field set to hex‘1B’, it shall send a link-level reject (LRJ) IU in response; if the channel provides control-unit emulation, the emulated control unit may send an LPE IU with the TYPE field set to hex‘1C’ in response (see 6.4.9,11.2.8, and 6.4.2 for additional information). IUs other than the ELP IU, LRJ IU, and LACK IU, which have the TYPE field set to hex‘1B’, shall be discarded by the channel and no response sent.

5.5.6 F_CTL Field

The F_CTL field controls sequences and exchanges. These bits are defined in FC-FS-4. FC-SB-6 has the following requirement on the relative offset present bit in the F_CTL field:

Relative Offset present: The relative offset present bit may be set to one for IUs of information category solicited data. For all other information categories, the relative offset present bit shall be set to zero (see 7.3.2.4, 7.4.2.2 and FC-FS-4). For command mode, the relative offset present bit may be set to one for all solicited data IUs with a data DIB. For transport mode, the relative offset present bit shall be set to one for all transport-data IUs.

Capability to perform sequence reassembly upon reception using SEQ_CNT (see FC-FS-4) in the absence of relative offset for command mode solicited data IUs with a data DIB shall be provided.

The settings of all other bits in the F_CTL field are defined in FC-FS-4.

5.5.7 SEQ_ID

The SEQ_ID field is defined by FC-FS-4.

5.5.8 DF_CTL

The DF_CTL field shall be set to zero.

5.5.9 SEQ_CNT

The SEQ_CNT field is defined by FC-FS-4.

NOTE 12 – When sending IUs, the use of continuously increasing sequence count is recommended. This increases the probability that errors caused by lost frames are detected.

5.5.10 OX_ID

The OX_ID field is defined by FC-FS-4.

5.5.11 RX_ID

The RX_ID field is defined by FC-FS-4.

5.5.12 Parameter Field

If the relative offset present bit is set to one in the F_CTL field of an IU sent with information category solicited data, then the parameter field shall contain the offset from the base address into which the data in a frame shall be stored. For command-mode data IUs with a data DIB, the base address for the solicited data information category is zero and refers to the first byte of each data IU. For transport-data IUs, the base address for the solicited data information category is zero and refers to the first byte of the first transport-data IU sent for the operation, and the relative offset shall have a value that is a multiple of 4 (i.e., each frame of each transport-data IU shall begin on a word boundary).

If the information category of an IU is not solicited data, the relative offset present bit shall be set to zero and the parameter field shall be ignored by the recipient (see 5.5.6 and see FC-FS-4).

6 Link-Level Functions

6.1 Link-Level Function Overview

FC-FS-4 link control, FC-FS-4 BLSs, FC-LS-3 ELSs, and FC-SB-6 link control provide the means by which the elements interconnected by the physical paths are brought to an operational state and maintained in an operational state (see clause 5, FC-FS-4 and FC-LS-3). The functions provided by FC-FS-4 BLSs, FC-LS-3 ELSs, and FC-SB-6 link control are collectively referred to as link-level functions. Link-level functions provided by FC-SB-6 link control are referred to as FC-SB-6 link-control functions. The link-level functions required for FC-SB-6 support are described in this clause.

6.2 FC-FS-4 Basic Link Services

6.2.1 Basic Link Services Overview

A brief overview of the Basic Link Services (BLS) commands required by FC-SB-6 is provided. See FC-FS-4 for detailed descriptions of these commands.

6.2.2 Abort Sequence

When an exchange is to be aborted, the abort sequence (ABTS) BLS shall be performed. For the conditions under which one or more exchanges are aborted, refer to clause 11 and 9.4.12. The settings of fields in the header of the ABTS BLS command are specified in FC-FS-4. When the recipient of an ABTS BLS command accepts it, the recipient shall send a basic accept (BA_ACC) response indicating that the entire exchange is being aborted and a recovery qualifier is to be established by setting the Last_Sequence bit in the F_CTL field to one, the Low SEQ_CNT field to hex’0000’, and the High SEQ_CNT field to hex’FFFF’. The recovery qualifier causes all frames for all sequences of the exchange to be discarded until the recovery qualifier is reinstated. After receiving the BA_ACC response and after waiting for an R_A_TOV timeout period, the sender of the ABTS BLS command shall send a reinstate recovery qualifier ELS command to allow the OX_ID and RX_ID combination to be reused (see 6.3.5 and FC-FS-4).

6.3 FC-LS-3 Extended Link Services

6.3.1 Extended Link-Services Overview

A brief overview of the Extended Link Services (ELS) commands required by FC-SB-6 is provided. See FC-LS-3 for detailed descriptions of these commands.

6.3.2 F_Port Login

Explicit F_Port login shall be performed by means of the F_Port login (FLOGI) ELS during the initialization process. See 7.3.2.2 for a definition of other aspects of explicit F_Port login specific to this standard, and see FC-LS-3 for general aspects of the explicit FLOGI ELS.

6.3.3 N_Port Login

Explicit N_Port login shall be performed by means of the N_Port login (PLOGI) ELS during the initialization process. When an N_Port has performed N_Port login with another N_Port, that N_Port is said to be logged in with the other N_Port.

When a PLOGI ELS command is received from an N_Port with which the recipient is currently logged in, the recipient shall perform implicit N_Port logout with the source N_Port and shall consider all open exchanges with the source N_Port to be abnormally terminated (see FC-FS-4 and 10.5.7) before accepting the PLOGI ELS command.

If the N_Port of a control unit completes explicit N_Port login with the N_Port of a channel at a time when one or more logical paths are indicated as being established to the channel, initiative to send a test initialization IU to that channel shall be generated (see 6.4.7). See 7.3.2.4 for a definition of other aspects of explicit N_Port login specific to this standard, and see FC-LS-3 for general aspects of the explicit PLOGI ELS.

6.3.4 N_Port Logout

N_Port logout may be performed either explicitly or implicitly. For the conditions under which implicit N_Port logout occurs, see FC-LS-3. When communication with the N_Port of a control unit is no longer required and no logical paths exist for which the channel is providing control-unit emulation, the N_Port of a channel shall initiate explicit N_Port logout by sending a LOGO ELS command and waiting for a response. The N_Port of a channel shall initiate explicit N_Port logout with the N_Port of a control unit if it receives an IU from the control unit and it is not configured to communicate with the control unit.

The N_Port of a control unit shall initiate explicit N_Port logout with the N_Port of a channel if the control unit receives an IU from a channel with which the N_Port of the control unit is not logged in. When this occurs, the received IU shall be discarded and no further action is taken by the N_Port of the control unit.

When the N_Port of a control unit accepts a LOGO ELS command from the N_Port of a channel to which logical paths exist, it shall reset its internal indicators for the logical paths established with the channel, and perform the equivalent of a system reset (9.4.4) for the affected logical paths.

NOTE 13 – FC-LS-3 requires that during the N_Port login procedure, other communication with the destination N_Port shall not be initiated or accepted. Therefore if a control unit has sent a PLOGI ELS to a channel and receives an IU from the channel before receiving a response to the PLOGI ELS, the received IU is discarded and no LOGO ELS is sent.

6.3.5 Reinstate Recovery Qualifier

The reinstate recovery qualifier (RRQ) ELS shall be performed in order to allow reuse of the OX_ID and RX_ID of an exchange which was aborted. When the sender of an ABTS BLS command receives a BA_ACC response, it shall send an RRQ ELS command after waiting an R_A_TOV timeout period. Other aspects of the Reinstate Recovery Qualifier ELS are given in FC-LS-3.

6.3.6 Registered State Change Notification

In a point-to-point configuration, the registered state change notification (RSCN) ELS shall be performed in order to notify the directly-attached N_Port of an event which has affected the state of the sending channel or control unit, provided that the attached N_Port has previously registered to receive state-change notifications (see 6.3.7). In a Fabric configuration, the RSCN ELS shall be performed in order to notify the Fabric Controller of an event which has affected the state of the sending N_Port. Additional aspects of the RSCN ELS including the format of the RSCN ELS request, the format of the response, and the conditions under which a Fabric Controller sends an RSCN ELS request are given in FC-LS-3.

The N_Port of a channel or a control unit may optionally perform the RSCN ELS when an event causes a change in existing logical paths or, for a control unit, in the ability to accept new logical paths. When the resources common to the N_Port of a channel are shared among channel images, the N_Port of that channel shall be capable of reporting a state change by performing the RSCN ELS.

When a control unit that was not previously capable of performing transport-mode operations becomes capable of performing transport-mode operations, the control unit shall send an RSCN ELS request, or cause an RSCN to be generated, as a result of the transport-mode capability change (see FC-LS-3 for conditions that cause an RSCN to be generated).

When a control unit that was not previously capable of supporting persistent IU pacing becomes capable of supporting persistent IU pacing, the control unit shall send an RSCN ELS request, or cause an RSCN to be generated, to all channel nodes with which the control unit currently has at least one logical channel path established.

The N_Port of a channel or control unit shall perform the RSCN ELS by sending the RSCN ELS request to the Fabric Controller if a Fabric is present, or to the N_Port of the attached control unit or channel if no Fabric is present. The address format byte (byte 1 of the Affected N_Port ID page of the RSCN ELS request) shall be set to zero to indicate that all three bytes of the N_Port ID are valid, and the affected N_Port ID shall be set to the N_Port ID of the sender. The sender of the RSCN ELS request shall not issue the request if it already signaled a condition which would cause the Fabric Controller, if present, to issue an RSCN ELS request on behalf of its N_Port ID (see FC-LS-3 for conditions that cause the Fabric Controller to issue an RSCN on behalf of an N_Port ID).

Once the N_Port of a channel or control unit recognizes a condition for which an RSCN ELS request is to be sent, initiative to send an RSCN ELS request is generated. Subsequent state changes in the sender shall not create initiative to send another RSCN ELS request if the initiative to send the first RSCN ELS request has not been discharged. Once the initiative to send the first RSCN ELS request has been discharged, a subsequent state change shall create a new initiative.

Initiative to send an RSCN ELS request shall be discharged when an accept (ACC) link service reply is received in response to the RSCN ELS request.

When an RSCN ELS request is accepted, the recipient shall check each affected N_Port ID page (see FC-LS-3). All three allowed values of the address format byte shall be supported. If an affected N_Port ID indicated in any affected N_Port ID page corresponds to an N_Port ID to which one or more logical paths are established, initiative to send a test initialization IU to that N_Port may be generated (see 6.4.7). If an affected N_Port ID corresponds to an N_Port ID to which no logical path is indicated as being established, initiative to send a test initialization (TIN) IU shall not be generated, and no further action is needed. However, at a channel, initiative to establish logical paths may be generated if its model-dependent configuration information indicates that a new control-unit image may have become available.

If, at a channel, an N_Port ID specified in an affected N_Port ID page is an N_Port ID corresponding to a control unit in the configuration for the channel, initiative to send an RNID ELS to the control unit shall be generated. Subsequent to the successful completion of the RNID ELS, a PRLI ELS may be issued to determine whether the control unit supports transport mode.

NOTE 14 – Sending the RSCN ELS request is not functionally equivalent to performing the FC-FS-4 online to offline primitive sequence protocol (see FC-FS-4 and 11.2.4). However, the RSCN ELS request may be used by a channel or control unit to report a state change that has affected logical paths. When multiple logical images share a common N_Port, RSCN is used in place of the online to offline primitive sequence protocol to report state changes that have not affected all of the logical images.

6.3.7 State-Change Registration

The state-change-registration (SCR) ELS shall be performed in order to register to receive RSCN ELS commands. General aspects of the SCR ELS, including the format of the SCR ELS command and response, are given in FC-LS-3. If a Fabric is present, the D_ID field in the FC-FS-4 header of the SCR

ELS command shall be set to the N_Port ID of the Fabric Controller. If no Fabric is present, the D_ID field shall be set to the N_Port ID of the attached channel or control unit. The registration function field in the payload of the SCR ELS command shall be set to hex'03' to request notifications to be sent for events detected by both the Fabric Controller and the affected N_Ports. After the SCR ELS command is accepted, registration remains in effect until N_Port logout occurs with the recipient of the SCR ELS command.

6.3.8 Query Security Attributes

The query security attributes (QSA) ELS shall be performed by a channel in a multi-switch Fabric configuration in order to determine the currently-enforced security attributes of the attached switch (see FC-LS-3 and 7.3.2.3)

NOTE 15 – See Annex A for a method by which a channel may determine the presence of a multi-switch Fabric configuration.

6.3.9 Request Node-Identification Data

6.3.9.1 Request Node-Identification Data Requests and Responses

The request-node-identification-data (RNID) ELS shall be performed in order to acquire the node identification data of the recipient (see FC-LS-3). When either the RNID ELS command or the RNID ACC ELS reply is sent, the node-identification data format field shall be set to hex'18'. In the RNID ACC ELS reply, the 32 byte node descriptor shown in figure 10, shall be returned in the specific node-identification data field. When the RNID ACC ELS is received, the common node-identification data, if present, may be ignored.

A channel shall establish initiative to send an RNID ELS:

a) as part of the channel initialization process described in 7.3.3;

b) to each control unit N_Port identified in an affected N_Port page for the RSCN that is in the configuration for the channel (see 6.3.6) when the channel has accepted an RSCN ELS; and

c) to each control unit N_Port in the configuration for the channel when the channel does not support persistent IU pacing and becomes capable of supporting the concurrent enablement of the persistent IU pacing function.

A control unit establishes initiative to send an RNID ELS in the following situations:

a) as part of the control unit initialization process described in 7.4.3; and

b) to a channel N_Port when the control unit receives an RNID ELS from the channel N_Port with which the control unit has established one or more logical paths.

6.3.9.2 Specific Node-Identification Data

For this standard, the specific node-identification data is referred to as the node descriptor. A node descriptor (ND) is a 32-byte field that describes a node. The ND consists of a 1 byte flags field, a 3 byte node parameters field, and a 28 byte node-ID.

The node-ID is composed of the following:

a) SDC ID: The first 26 bytes of the node-ID identify the Self-Describing Component (SDC) containing the interface that determines the node. The SDC ID shall correspond to the information provided on a serial-number plate attached to the external surface of the structure containing the SDC; and

b) Interface ID (Tag): The last two bytes of the node-ID shall contain an interface identifier (ID) that uniquely identifies the physical location of the associated SDC interface.

Node-IDs with the same SDC ID shall not use the same interface ID.

Collectively, the 25 bytes of information contained in word 1, bytes 0 and 1 of word 2, bytes 1-3 of word 3, and words 4-7 of the node descriptor shall provide a vendor specific node identifier.

A node descriptor shall have the format given in figure 10.

<table>
<tr><th>Word</th><th colspan="4"></th></tr>
<tr><td>0</td><td>Flags</td><td colspan="3">Node Parameters</td></tr>
<tr><td>1</td><td colspan="4">Type Number</td></tr>
<tr><td>2</td><td colspan="2">Type Number (continued)</td><td colspan="2">Model Number</td></tr>
<tr><td>3</td><td>Model No. (cont.)</td><td colspan="3">Manufacturer</td></tr>
<tr><td>4</td><td colspan="2">Plant of Manufacture</td><td colspan="2">Sequence Number</td></tr>
<tr><td>5</td><td colspan="4">Sequence Number (continued)</td></tr>
<tr><td>6</td><td colspan="4">Sequence Number (continued)</td></tr>
<tr><td>7</td><td colspan="2">Sequence No. (continued)</td><td colspan="2">Tag</td></tr>
<tr><td>Bit</td><td colspan="2">0</td><td>16</td><td>31</td></tr>
</table>

Figure 10 – Contents of the Node Descriptor

Flags: Byte 0 of word 0 describes the manner in which selected fields of the node descriptor are to be interpreted. The meaning of bits 0-7 is as follows:

<u>*Bit*</u> <u>*Meaning*</u>

0-2 Node-ID validity. Bits 0-2 contain a code that describes the validity of bits 3-7 of the flags field, the node parameters field, and the node-ID contained in words 1-7. The codes and their meanings are as follows:

Value	*Meaning*
0	Bits 3-7 of the flags field, the node parameters field, and the node-ID are valid.
1	Bits 3-7 of the flags field, the node parameters field, and the node-ID are valid, however, they may not be current. This value shall be used when the SDC has obtained the requested node ID but subsequently has observed some event (e.g., as the loss of signal on a link) which may have resulted in a configuration change and the SDC has been unable to obtain the node ID again.
2	Node-ID is not valid. The SDC is unable to obtain the requested node-ID. Except for the node-ID-validity field, the contents of the node descriptor shall have no meaning.
3-7	Reserved.

When the N_Port of a channel or control unit receives an RNID ELS command, the channel or control unit shall determine the node-ID validity and set the appropriate node-ID validity code in the flag field of the ND. If, while obtaining its node-ID, an error is detected or it is determined for some other reason that the node-descriptor information is suspect (i.e., not valid), then the node-ID validity code shall be set to the value 2 in the flag field of the node descriptor being sent in the accept response.

In the accept response to the RNID ELS command, a node-ID validity code of either 0 or 2 in the flag field of the node descriptor shall be indicated. Since a node descriptor is either *valid* or *not valid*, a node-ID validity code of 1 shall never be sent in the accept response to the RNID command.

3 Node type. When zero, bit 3 shall specify that the node described by this node descriptor is a device-type node; when one, this bit shall specify that the node described by this node descriptor is a central-processor-complex-type (CPC-type) node.

4-7 Reserved.

Node Parameters: Bytes 1-3 of word 0 shall contain additional information about the node.

When bit 3 of the flags field is zero, indicating that this is a device-type node, the contents of bytes 1-3 of word 0 shall be as follows:

Byte	*Description*
1	*Bits 0-2* of byte 1 contain a code that shall specify the interface protocol type of the interface identified by the node descriptor. The codes and their meanings are as follows:

Value	*Meaning*
0	Reserved.
1	FC-SB
2	FC-4s other than FC-SB, such as FCP
3	FC-SB and other FC-4s, such as FCP
4	FC-4 support not specified (this shall not be used if another value applies)
5-6	Reserved
7	vendor specific

A node which supports the requirements of this standard shall use code 1 or 3.

Bit 3 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 3 of byte 1 indicates the following:

a) When the bit is one, the node supports FC-SB process login using the process login (PRLI) ELS; or

b) When the bit is zero, the node does not support FC-SB process login.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 3 of byte 1 is reserved and set to zero.

Bit 4 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 4 of byte 1 indicates the following:

a) When the bit is one, the node supports the TINC and TINCR functions; or

b) When the bit is zero, the node may not support the TINC and TINCR functions.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 4 of byte 1 is reserved and set to zero.

Bit 5 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for an FC-4 interface protocol, bit 5 of byte 1 indicates the Fibre Channel physical interface type as follows:

a) When the bit is one, the physical interface is Fibre Channel over Ethernet (FCoE); or

b) When the bit is zero, the physical interface is the Fibre Channel native interface.

When bits 0-2 of byte 1 do not indicate support for an FC-4 interface protocol, bit 5 of byte 1 is reserved and set to zero.

Bit 6 of byte 1 is reserved.

Bit 7 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 7 of byte 1 indicates the Fibre Channel physical interface type as follows:

a) When the bit is one, the node supports concurrent enablement of the persistent IU pacing function; or

b) When the bit is zero, the node does not support concurrent enablement of the persistent IU pacing function.

When a channel that supports concurrent enablement of the persistent IU pacing function receives a node descriptor from a control unit with bit 7 of byte 1 equal to one, the channel enables persistent IU pacing for all currently established logical paths with the control unit. For logical paths that are established subsequent to the processing of the node descriptor, the persistent IU pacing bit in the ELP/LPE IU optional features field is used to enable or disable persistent IU pacing.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 7 of byte 1 is reserved and set to zero.

2 Class. Byte 2 of word 0 shall contain a code that specifies the class to which the device belongs. The codes and their meanings are as follows:

Value	*Meaning*
0	Unspecified class

1	Direct access storage (DASD)
2	Magnetic tape
3	Unit record (input)
4	Unit record (output)
5	Printer
6	Communications controller
7	Terminal (full screen)
8	Terminal (line mode)
9	Reserved
10	Fabric
11-255	Reserved

3 Byte 3 shall contain zero, except in the following case:

When code 10 is specified in the class field byte 3 shall contain the area field (bits 15 - 8) of the Port address identifier of the associated switch interface.

When bit 3 of the flags field is one, indicating that this is a CPC-type node, the contents of bytes 1-3 of word 0 shall be as follows:

<u>*Byte*</u> <u>*Description*</u>

1 *Type*. When the class field contains a value other than 1, byte 1 of word 0 shall be reserved and set to zero.

When the class field contains a value of 1, byte 1 of word 0 shall be defined as follows:

Bits 0-2 of byte 1 contain a code that specifies the interface protocol type of the interface identified by the node descriptor. The codes and their meanings shall be as follows:

Value	*Meaning*
0	Reserved.
1	FC-SB
2	FC-4s other than FC-SB, such as FCP
3	FC-SB and other FC-4s, such as FCP
4	FC-4 support not specified (this shall not be used if another value applies)
5-6	Reserved
7	vendor specific

Bit 3 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 3 of byte 1 indicates the following:

a) When the bit is one, the node supports FC-SB process login using the process login (PRLI) ELS; or

b) When the bit is zero, the node does not support FC-SB process login.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 3 of byte 1 is reserved and shall be set to zero.

Bit 4 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 4 of byte 1 indicates the following:

a) When the bit is one, the node supports the TINC and TINCR functions; or

b) When the bit is zero, the node may not support the TINC and TINCR functions.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 4 of byte 1 is reserved and shall be set to zero.

Bit 5 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for an FC-4 interface protocol, bit 5 of byte 1 indicates the Fibre Channel physical interface type as follows:

a) When the bit is one, the physical interface is Fibre Channel over Ethernet (FCoE); or

b) When the bit is zero, the physical interface is the Fibre Channel native interface.

When bits 0-2 of byte 1 do not indicate support for an FC-4 interface protocol, bit 5 of byte 1 is reserved and shall be set to zero.

Bit 6 of byte 1 is reserved.

Bit 7 of byte 1 is defined as follows:

When bits 0-2 of byte 1 indicate support for the FC-SB interface protocol, bit 7 of byte 1 indicates the following:

a) When the bit is one, the node supports concurrent enablement of the persistent IU pacing function; or

b) When the bit is zero, the node does not support concurrent enablement of the persistent IU pacing function.

When a channel that supports concurrent enablement of the persistent IU pacing function receives a node descriptor from a control unit with bit 7 of byte 1 equal to one, the channel enables persistent IU pacing for all currently established logical paths with the control unit. For logical paths that are established subsequent to the processing of the node descriptor, the persistent IU pacing bit in the ELP/LPE IU optional features field is used to enable or disable persistent IU pacing.

When bits 0-2 of byte 1 do not indicate support for the FC-SB interface protocol, bit 7 of byte 1 is reserved and shall be set to zero.

2 *Class*. Byte 2 of word 0 shall contain a code that specifies the class to which the interface belongs. The codes and their meanings shall be as follows:

Value	*Meaning*
0	Unspecified class

1	Channel path, not CTC capable
2-6	Reserved
7	FC-SB for the channel-to-channel adapter (FC-SB CTCA)
8	Emulated control unit support only
9-255	Reserved

3 *Identification*. Byte 3 of word 0 shall contain the CHPID of the channel path that contains the specified interface.

The contents of the following five fields shall correspond to the information provided on a serial-number plate attached to the external surface of the SDC. The ASCII character code used in these fields shall be ASCII/Latin 1.

Type Number: Word 1 and bytes 0-1 of word 2 shall contain the six-character (0-9) ASCII type number of the SDC. The type number shall be right justified with leading ASCII zeros if necessary.

Model Number: Bytes 2-3 of word 2 and byte 0 of word 3 shall contain, if applicable, the three-character (0-9 or uppercase A-Z) ASCII model number of the SDC. The model number shall be right justified with leading ASCII zeros if necessary.

Manufacturer: Bytes 1-3 of word 3 shall contain a three-character (0-9 or uppercase A-Z) ASCII code that identifies the manufacturer of the SDC.

Plant of Manufacture: Bytes 0-1 of word 4 shall contain a two-character (0-9 or uppercase A-Z) ASCII plant code that identifies the plant of manufacture for the SDC.

Sequence Number: Bytes 2-3 of word 4, words 5-6, and bytes 0-1 of word 7 shall contain the 12-character (0-9 or uppercase A-Z) ASCII sequence number of the SDC. The sequence number shall be right justified with leading ASCII zeros if necessary.

A serial number shall consist of the concatenation of the plant-of-manufacture designation with the sequence-number designation.

Tag: Bytes 2-3 of word 7 shall contain the physical identifier for the SDC interface that is identified by the preceding 26 bytes of the node descriptor.

6.3.10 Registered Link-Incident Record

6.3.10.1 Registered Link-Incident Record Requests and Responses

The registered link-incident record (RLIR) ELS provides the means to send a link-incident record to the N_Port of a channel. General aspects of the RLIR ELS, including the format of the RLIR ELS command and ACC reply, are given in FC-LS-3. When the RLIR ELS command is sent, the link-incident record format field shall be set to hex’18’. When the RLIR ELS command is received, the link-incident descriptor may be ignored and the common link-incident record, if present, may be ignored. The specific link-incident record field shall contain a 100 byte FC-SB-6 specific link-incident record of the format described in 6.3.10.2.

Control units shall use the link-incident reporting procedure to report link-incidents.

6.3.10.2 Specific Link-Incident Record for FC-SB-6

The specific link-incident record for FC-SB-6 is shown in table 2.

Table 2 – Specific Link-Incident Record for FC-SB-6

Byte	Contents		
0-3	IQ	IC	Reserved
4-35	Incident Node Descriptor		
36-67	Attached Node Descriptor		
68-99	Incident Specific Information		

Incident Qualifier (IQ): Byte 0 shall describe the manner in which the contents of the link-incident record shall be interpreted. The meaning of bits 0-7 is as follows:

Bit	*Meaning*
0	Reserved
1	Reserved.
2	Switch: When one, bit 2 shall indicate that the incident node, identified by the incident-node descriptor, is a switch node. When zero, bit 2 shall indicate that the incident node is not a switch node.
3	Reserved.
4-5	Bits 4 and 5 constitute a two-bit code which shall identify the reporting class for the link-incident. The codes and their meanings are as follows:

Value	*Meaning*
0	Informational report: All link-incidents reported with incident code bit 0 set to one shall use a reporting class value of zero.
1	Link degraded but operational: Link-incidents reported with incident -code bit 0 set to zero shall use reporting class 1 if the link associated with the incident node is not in a link-failure or offline state as a result of the event which generated the link-incident record.
2	Link not operational: Link-incidents reported with incident code bit 0 set to zero shall use reporting class 2 if the link associated with the incident node is in a Link-Failure or Offline state as a result of the event which generated the link-incident record.
3	Reserved.

6 Subassembly type: When one, bit 6 shall specify that the type of subassembly used for the node that is the subject of this link-incident record is laser. When zero, bit 6 shall specify that the type of subassembly used for the node that is the subject of this link-incident record is not laser.

7 FRU identification: When one, bit 7 shall specify that the incident-specific-information field is in a format that provides field-replaceable-unit (FRU) identification. When zero, bit 7 shall specify that the incident-specific-information field is vendor specific.

Incident Code (IC): Byte 1 shall contain the incident code which describes the incident that was observed by the incident node.

Bit	Meaning
0	Bit 0 of the incident code shall indicate whether the link-incident record is a primary or secondary report of the link-incident. When bit 0 is set to zero, the link-incident record shall be a primary report. When bit 0 is set to one, the link-incident record shall be a secondary report.
1-7	Bits 1-7 of the incident code shall contain a value that specifies the type of incident which was observed. The values that may be specified are as follows:

Value	Meaning
0	Reserved.
1	Implicit Incident: A condition which has been caused by an event known to have occurred within the incident node has been recognized by the incident node. The condition affects the attached link in such a way that it may cause a link-incident to be recognized by the attached node.
2	Bit-error-rate Threshold Exceeded: The number of code violation errors recognized by the incident node has exceeded a threshold (see FC-FS-4).
3	FC-FS-4 Link Failure - Loss of Signal or Loss of Synchronization: A loss of synchronization condition has been recognized by the incident node, and it persisted for more than the R_T_TOV timeout period. A loss of signal condition has been recognized by the incident node (see FC-FS-4).
4	FC-FS-4 Link Failure - Not-Operational Primitive Sequence (NOS) Recognized: The NOS primitive sequence has been recognized by the incident node (see FC-FS-4).
5	FC-FS-4 Link Failure - Primitive Sequence Timeout: The incident node has recognized either a link reset protocol timeout (see FC-FS-4) or a timeout when timing for the appropriate response while in the NOS Receive state and after the NOS is no longer recognized (i.e., Event Timeout, see FC-FS-4).
6	FC-FS-4 Link Failure - Invalid Primitive Sequence for Port State: Either a link reset (LR) or link reset response (LRR) primitive sequence was recognized by the incident node while in the wait-for-OLS state (see FC-FS-4). A link-incident record shall be generated and reported for an FC-FS-4 link-failure condition only if the FC-FS-4 link-failure condition persists for longer than SB_TOV.
7-127	Reserved.

Incident-Node Descriptor: Bytes 4-35 shall contain the node descriptor of the incident node. The contents of a node descriptor are described in 6.3.9.2.

Attached-Node Descriptor: Bytes 36-67 shall contain the node descriptor of the node attached to the incident node at the time the link-incident was detected. The contents of a node descriptor are described in 6.3.9.2.

Incident-Specific Information: When bit 7 of the incident-qualifier field is set to zero, bytes 68-99 shall contain node-dependent incident information, which may provide additional information related to the incident.

When bit 7 of the incident-qualifier field is set to one, bytes 68-99 shall contain field-replaceable-unit (FRU) identification information.

When the incident-specific-information field contains FRU identification information, the format of the incident-specific information is shown in table 3:

Table 3 – Incident-Specific Information

Bytes	Contents	
68 - 69	FRU Flags	Reserved
70 - 81	First FRU Callout (12 Bytes)	
82 - 93	Second FRU Callout Model-Dependent Information (12 Bytes)	
94 - 99	Reserved	

Byte 68 shall contain the FRU-flags field. The meaning of bits 0-7 shall be as follows:

Bit *Meaning*

0 Reserved.

1 Format bit; FRU-callout-field format:

Value	*Meaning*
0	FRU-part-number format
1	FRU-code format

2-5 Reserved.

6-7 Validity code for FRU-callout fields:

Value	*Meaning*
00	Reserved.
01	First-FRU-callout field valid; Second-FRU-callout field contains 12 bytes of model-dependent data.
10	First-FRU-callout and second-FRU-callout fields valid.
11	Reserved.

Byte 69 shall be reserved and set to zero.

Bytes 70-81 shall contain the first-FRU-callout identification information.

Bytes 82-93 shall contain either the second-FRU-callout identification information or 12 bytes of model-dependent information, depending on the value of bits 6-7 of the FRU-flags field.

Bytes 94-99 shall be reserved and set to zeros.

The format of a valid FRU-callout field depends on the value of bit 1 (format bit) of the FRU-flags field within the same link-incident record.

When the format bit is set to zero, the FRU-callout field shall be in ASCII, right justified, with either leading blanks (hex’20’) or leading ASCII zeros (hex’30’).

When the format bit is set to one, the FRU-callout field shall be in hexadecimal, right justified, with leading zeros (hex’00’).

6.3.11 Link-Incident-Record Registration

The link-incident-record registration (LIRR) ELS provides the means for the N_Port of a channel to register to receive link-incident records from a control unit. General aspects of the LIRR ELS including the format of the LIRR ELS command and ACC response are given in FC-LS-3.

The N_Port of a channel shall send an LIRR command during the initialization procedure (see clause 7). The registration function field shall be set to hex’01’ to indicate the set registration-conditionally receive function, and the link-incident-record registration type field shall be set to hex’18’. Although it is possible to clear registration for link-incident records by setting the registration function field to hex’FF’, the sender shall never set the registration function field to this value.

6.3.12 Read Link Error Status Block

If a control unit supports error code transfer in Purge Path IUs, it shall use the Read Link Error Status Block (RLS) ELS request to attempt to obtain the link error status block (LESB) of the attached F_Port (see 8.11.2.14). General aspects of the RLS ELS, including the format of the RLS ELS command and response, are given in FC-LS-3.

6.3.13 Registered Fabric Change Notification

If a channel receives the Registered Fabric Change Notification (RFCN) ELS request indicating that a security attribute is changing, it shall perform the channel login and security attribute determination link-initialization procedure. See 7.3.2. General aspects of the RFCN ELS, including the format of the RFCN ELS command and response, are given in FC-LS-3.

6.3.14 Process Login

6.3.14.1 Process Login Overview

A channel that supports the process login (PRLI) ELS and transport-mode operations shall send a PRLI request to each control unit in its configuration that also supports the process login ELS to determine whether the control unit supports transport-mode operations. General aspects of the PRLI ELS, including the format of the PRLI ELS request and response, are given in FC-LS-3 with specific settings for FC-SB-6 defined in 6.3.14. The PRLI request is sent during channel initialization (see 7.3) prior to establishing logical paths and may also be sent as the result of a process logout.

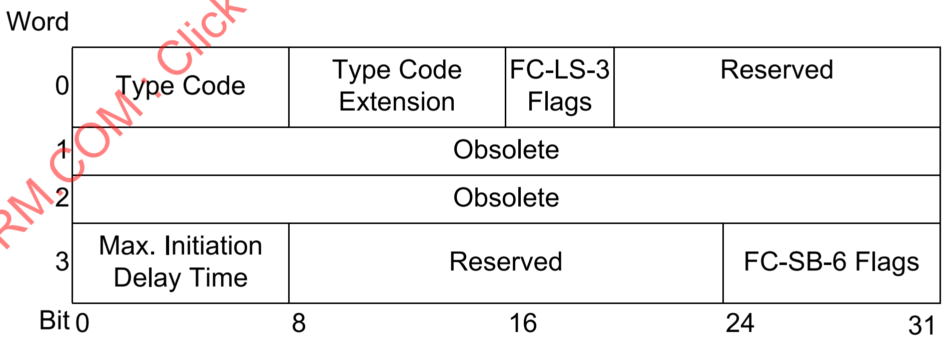
The PRLI ELS is used to exchange process login service parameters between a channel and control unit and is not used to establish logical image pairs or to set process associators. A PRLI ELS shall not be sent to a control unit that does not support FC-SB-6 process login. Support for the FC-SB-6 process login is indicated in the node descriptor for the control unit (see 6.3.9). A PRLI ELS request may be sent by a channel to a control unit when logical paths are established with the control unit and applies to all the established logical paths.

If a control unit receives a PRLI ELS request that would modify the service parameters for a process login already in effect with the channel, the control unit responds to the PRLI ELS request with a link service reject (LS_RJT). The LS_RJT response is sent with reason code hex’09’, “unable to perform command request at this time”. After the control unit sends the LS_RJT, the control unit shall send a process logout (PRLO) ELS request to the channel to allow a retry of the PRLI ELS request from the channel to be successfully performed.

If the channel receives an LS_ACC response for a PRLI request that would modify the service parameters for a process login already in effect with the control unit, the channel performs a PRLO ELS with the control unit. The channel establishes initiative to send a PRLI ELS request to the control unit after completion of the process logout.

6.3.14.2 PRLI Request Service Parameter Page

The content of the FC-SB-6 PRLI request service parameter page is described in figure 11.



Word	Bit 0	8	16	24–31
0	Type Code	Type Code Extension	FC-LS-3 Flags	Reserved
1	Obsolete			
2	Obsolete			
3	Max. Initiation Delay Time	Reserved		FC-SB-6 Flags

Figure 11 – Service Parameter Page for a PRLI Request

Bytes 1 and 2 of word 3 are reserved and shall be set to zero.

Type Code: Byte 0 of word 0 identifies the FC-4 protocol and shall be to set to hex’1B’ to indicate the FC-SB-6 protocol.

Type Code Extension: Byte 1 of word 0 shall be set to zero.

FC-LS-3 Flags: Bits 16-19 of word 0 are defined by FC-LS-3 and are set as follows:

Bit	*Meaning*
16	Obsolete.
17	Obsolete.
18	Establish Image Pair. Bit 18 of word 0 shall be set to zero by the channel and ignored by the control unit.
19	Reserved. Bit 19 of word 0 shall be set to zero by the channel and ignored by the control unit.

Maximum Initiation Delay Time: Byte 0 of word 3 contains a binary integer that specifies in units of seconds the maximum value that the control unit can set as the initiation-delay time in a PRLO request (see 6.3.15.4). This value is generally based on the amount of time that the channel can hold off initiating new I/O operations without disrupting the system.

FC-SB Flags: Byte 3 of word 3 contains flags defined by this standard as follows:

Bit	*Meaning*
0	Transport Mode Supported. When bit 0 is set to zero, the channel does not support transport-mode operations. When bit 0 is set to one, the channel supports transport-mode operations.
1	Reserved.
2	DCW Incorrect Length Facility Supported. When bit 2 is set to one, the DCW-incorrect-length facility (see 9.3.3.3) is supported by the channel. When bit 2 is set to zero, the DCW-incorrect-length facility is not supported by the channel. When the transport-mode-supported bit (bit 0) is equal to zero, bit 2 shall be set to zero by the channel and shall be ignored by the control unit.
3	Transport Mode Command Retry Supported. When bit 3 is set to one, transport-mode command retry (see 9.5.2.2) is supported by the channel. When bit 3 is set to zero, transport-mode command retry is not supported by the channel. When the transport-mode-supported bit (bit 0) is equal to zero, bit 3 shall be set to zero by the channel and shall be ignored by the control unit.
4	First Transfer Buffer Credits Supported. When bit 4 is set to one, the use of first-transfer-buffer credits (see 9.3.2.2.2) is supported by the channel. When bit 4 is set to zero, the use of first-transfer-buffer credits is not supported by the channel. When the first-transfer- ready-disabled-supported bit (bit 7) is zero, bit 4 shall be set to zero by the channel and shall be ignored by the control unit.
5	Bidirectional Data Transfer Supported. When bit 5 is set to one, bidirectional data transfer (see 9.3.2.4) is supported by the channel. When bit 5 is set to zero, bidirectional data transfer is not supported by the channel. When the transport-mode-supported bit (bit 0) is equal to zero, bit 5 shall be set to zero by the channel and shall be ignored by the control unit.
6	Reserved.

7 First Transfer Ready Disabled Supported. When bit 7 is set to one, first-transfer-ready-disabled operation (see 9.3.2.2) is supported by the channel. When bit 7 is set to zero, first-transfer-ready-disabled operation is not supported by the channel. When the transport-mode-supported bit (bit 0) is equal to zero, bit 7 shall be set to zero by the channel and shall be ignored by the control unit.

6.3.14.3 PRLI Accept Service Parameter Page

The content of the FC-SB-6 PRLI accept service parameter page is described in figure 12.

Word					
0	Type Code	Type Code Extension	FC-LS-3 Flags	Resp Code	Reserved
1	Obsolete				
2	Obsolete				
3	First Burst Size or First Transfer Buffer Size		Reserved		FC-SB-6 Flags
Bit	0	8	16	20	24 31

Figure 12 – Service Parameter Response Page for a PRLI LS_ACC

Byte 3 of word 0 and byte 2 of word 3 are reserved and shall be set to zero.

Type Code: Byte 0 of word 0 identifies the FC-4 protocol and shall be to set to hex'1B' to indicate the FC-SB-6 protocol.

Type Code Extension: Byte 1 of word 0 shall be set to zero.

FC-LS-3 Flags: Bits 16-19 of word 0 are defined by FC-LS-3 and are set as follows:

Bit	Meaning
16	Obsolete.
17	Obsolete.
18	Image Pair Established. Bit 18 of word 0 shall be set to zero by the control unit and ignored by the channel.
19	Reserved. Bit 19 of word 0 shall be set to zero by the control unit and ignored by the channel.

Response Code: Bits 20-23 of word 0 contain a binary integer indicating the result of the PRLI Request. The meanings of the response codes values are defined by FC-LS-3.

First Burst Size/First Transfer Buffer Size: If the use of first-transfer-buffer credits is not supported by the control unit or the channel, bytes 0-1 of word 3 shall contain the first-burst size. When non-zero, the first-burst size is an unsigned 16-bit binary integer that specifies the maximum amount of data in units of 4k bytes that are allowed to be sent in the first transport-data IU for a write data transfer when first-transfer-ready disabled is in effect for a transport-mode operation (see 9.3.2.2). A value of zero shall indicate that there is no first-burst size limit specified by the control unit.

If the use of first-transfer-buffer credits (see 9.3.2.2.2) is supported by both the channel and control unit, bytes 0-1 of word 3 shall contain the first-transfer-buffer size. The first-transfer-buffer size is an unsigned 16-bit binary integer that specifies in units of 4k bytes the size of each first-transfer buffer. A first-transfer-buffer-size equal to zero indicates that there is no buffer space allocated by the control unit for a first write transport-data IU when first-data-transfer-ready disabled is in effect (see 9.3.2.2.2).

FC-SB Flags: Byte 3 of word 3 contains flags defined by this standard as follows:

Bit	*Meaning*
0	Transport Mode Supported. When bit 0 is set to zero, the control unit does not support transport-mode operations. When bit 0 is set to one, the control unit supports transport-mode operations.
1	Reserved.
2	DCW Incorrect Length Facility Supported. When bit 2 is set to one, the DCW-incorrect-length facility (see 9.3.3.3) is supported by the control unit. When bit 2 is set to zero, the DCW-incorrect-length facility is not supported by the control unit. When the transport-mode-supported bit (bit 0) is equal to zero, bit 2 shall be set to zero by the control unit and shall be ignored by the channel.
3	Transport Mode Command Retry Supported. When bit 3 is set to one, transport-mode command retry (see 9.5.2.2) is supported by the control unit. When bit 3 is set to zero, transport-mode command retry is not supported by the control unit. When the transport-mode-supported bit (bit 0) is equal to zero, bit 3 shall be set to zero by the control unit and shall be ignored by the channel.
4	First Transfer Buffer Credits Supported. When bit 4 is set to one, the use of first-transfer-buffer credits (see 9.3.2.2.2) is supported by the control unit. When bit 4 is set to zero, the use of first-transfer-buffer credits is not supported by the control unit. When the first-transfer-ready-disabled-supported bit (bit 7) is zero, bit 4 shall be set to zero by the control unit and shall be ignored by the channel.
5	Bidirectional Data Transfer Supported. When bit 5 is set to one, bidirectional data transfer (see 9.3.2.4) is supported by the control unit. When bit 5 is set to zero, bidirectional data transfer is not supported by the control unit. When the transport-mode-supported bit (bit 0) is equal to zero, bit 5 shall be set to zero by the control unit and shall be ignored by the channel.
6	Reserved.
7	First Transfer Ready Disabled Supported. When bit 7 is set to one, first-transfer-ready-disabled operation (see 9.3.2.2) is supported by the control unit. When bit 7 is set to zero, first-transfer-ready-disabled operation is not supported by the control unit. When the transport-mode-supported bit (bit 0) is equal to zero, bit 7 shall be set to zero by the control unit and shall be ignored by the channel.

6.3.15 Process Logout

6.3.15.1 Process Logout Overview

A channel or control unit that has successfully performed process login uses the Process Logout (PRLO) ELS to request that a process logout be performed. A process logout does not cause logical paths between a channel and control to be removed. General aspects of the PRLO ELS, including the format of the PRLO ELS request and response, are given in FC-LS-3 with specific definitions for this standard defined in 6.3.15.

Process logout shall terminate transport-mode operation between a channel and control unit. When the process logout procedure completes, the channel shall not initiate transport-mode operations to the control unit until a subsequent PRLI is performed to re-establish transport-mode operating parameters.

6.3.15.2 Channel Process Logout Procedure

When a channel determines that it will issue a PRLO ELS request, it shall not send transport-mode or command-mode initiation IUs to the control unit for a period equal to the exchange quiesce timeout value (EQ_TOV - see 10.2.10) prior to sending the PRLO ELS request. During this period, the channel continues processing open exchanges to attempt to reduce the number of open exchanges with the control unit when the PRLO ELS request is sent.

When a channel sends a PRLO ELS request, it shall not send transport-mode or command-mode initiation IUs to the control unit until it completes process logout. If the channel receives an LS_RJT to the PRLO ELS request, the channel shall retry the request a minimum of one time. When the channel receives an LS_ACC response to the PRLO ELS request or exhausts retry attempts, the channel shall abort all open transport exchanges with the control unit by invoking ABTS to complete the process logout function at the channel.

When the channel receives a PRLO ELS request from a control unit, the channel shall not send transport-mode or command-mode initiation IUs to the control unit for a period equal to PL_TOV. During the PL_TOV period, the channel shall continue processing any open exchanges. When PL_TOV expires, the channel shall respond to the PRLO request with an LS_ACC response and shall abort all open transport exchanges with the control unit by invoking ABTS to complete the process logout at the channel.

Subsequent to the completion of a process logout, the channel shall successfully complete an RNID request to the control unit before initiating new I/O operations to the control unit. The channel shall delay for an amount of time equal to the initiation-delay time specified in the PRLO ELS request before sending the RNID ELS request. During the initiation-delay time interval, the channel processes status presented by the control unit. If, during the delay interval, the channel determines that a TIN command is required, initiative to perform the TIN command is established after the delay interval expires or the delay interval is otherwise terminated.

If an LS_ACC response is received for the RNID request, and the control unit indicates support for FC-SB-6 process login, the channel shall send a PRLI ELS request (see 6.3.14) to the control unit if the channel supports FC-SB-6 process login. If the PRLI response from the control unit indicates support for transport-mode operations and the channel supports transport-mode operations, the channel shall resume operation and support transport-mode operations; otherwise, the channel shall resume operation without transport-mode support until a subsequent process login is performed that enables transport-mode operation.

If an LS_ACC response is not received for the RNID request, the channel establishes initiative to perform the TIN command and maintains initiative to perform the RNID. If an RNID request does not complete successfully prior to the completion of the TIN or LP_TOV, the channel shall reset internal path-established indicators for all logical paths to the control unit.

6.3.15.3 Control Unit Process Logout Procedure

When a control unit determines that it shall issue a PRLO request to a channel, it shall respond to initiation IUs containing a device-level function with control-unit busy status for a period equal to EQ_TOV prior to issuing the PRLO request. During this delay period, the control unit continues processing open exchanges to attempt to reduce the number of open exchanges with the channel before the PRLO request is issued. When the control unit is able to accept initiation IUs containing a device-level function, control-unit-end status shall be sent to each logical path that accepted a status byte with the control-unit-busy indication (see 9.4.10).

The control unit shall issue a PRLO request when it has a process login in effect with a channel when it needs to modify the existing process login service parameters. Modified service parameters may be sent by the control unit to the channel in an LS_ACC response to a subsequent PRLI ELS request from the channel. The control unit shall also issue a PRLO request in the following circumstances:

a) When it receives a transport-command IU from a channel for which a process login is not in effect; or

b) When it receives a PRLI ELS request from a channel that would modify service parameters for a process login already in effect with the channel.

If the control unit receives an LS_RJT to the PRLO ELS request, the control unit may retry the PRLO ELS. When the control unit receives an LS_ACC response to the PRLO ELS request or exhausts any retry attempts, the control unit shall abort all open transport exchanges with the channel by invoking ABTS to complete the process logout at the control unit. The control unit shall resume operation without transport-mode support until a subsequent process login is performed that enables transport-mode operation.

When a control unit receives a PRLO ELS request, it shall respond to all transport-mode and command-mode initiation IUs with control unit busy status for PL_TOV. During PL_TOV, the control unit shall continue processing any open exchanges. After PL_TOV expires, the control unit shall respond to the PRLO ELS request with an LS_ACC response and shall abort all open transport exchanges with the channel by invoking ABTS to complete the process logout at the control unit.

After sending an LS_ACC for a PRLO ELS request that contained a non-zero initiation-delay time, the control unit shall not send an ELS request or initiate an exchange with the channel until one of the following occur:

a) the initiation-delay time specified in the PRLO ELS request has expired;

b) the control unit receives and accepts an ELS request from the channel; or

c) the control unit receives an initiation IU from the channel.

The control unit shall operate without transport-mode support until a process login is performed that enables transport-mode operation.

6.3.15.4 PRLO Request Logout Parameter Page

The content of the FC-SB-6 PRLO request logout parameter page is shown in figure 13.

Word				
0	Type Code	Reserved	FC-LS-3 Flags	
1	Obsolete			
2	Obsolete			
3	Initiation Delay Time	Reserved		
Bit	0	8	16	31

Figure 13 – Logout Parameter Page for a PRLO request

Byte 1 of word 0 and bytes 1-3 of word 3 are reserved and shall be set to zero.

Type Code: Byte 0 of word 0 identifies the FC-4 protocol and shall be to set to hex’1B’ to indicate the FC-SB-6 protocol.

FC-LS-3 Flags: Bits 16-31 of word 0 are defined by FC-LS-3 and are set as follows:

Bit	Meaning
16	Obsolete.
17	Obsolete.
18-31	Reserved. Shall be set to zeros.

Initiation Delay Time: Byte 0 of word 3 contains a binary integer that specifies in units of seconds the minimum amount of time that the recipient of the PRLO ELS request shall delay before sending an ELS or initiating an exchange or exchange pair with the sender of the PRLO ELS request. When the PRLO is being sent as the result of transport-command IU from a channel for which a process login is not currently established, this value is set to zero (see 6.3.15.2 and 6.3.15.3).

6.3.15.5 PRLO Accept Logout Parameter Page

.The content of the FC-SB-6 PRLO accept logout parameter page is shown in figure 14.

Word						
0	Type Code	Reserved	FC-LS-3 Flags	Resp Code	Reserved	
1	Obsolete					
2	Obsolete					
3	Reserved					
Bit	0	8	16	20	24	31

Figure 14 – Logout Parameter Page for a PRLO LS_ACC

Bytes 1 and 3 of word 0 and word 3 are reserved and shall be set to zero.

Type Code: Byte 0 of word 0 identifies the FC-4 protocol and shall be to set to hex'1B' to indicate the FC-SB-6 protocol.

FC-LS-3 Flags: Bits 16-19 of word 0 are defined by FC-LS-3 and are set as follows:

Bit	*Meaning*
16	Obsolete.
17	Obsolete.
18-19	Reserved. Shall be set to zeros.

Response Code: Bits 20-23 of word 0 contain a binary integer indicating the result of the PRLO Request. The meanings of the response codes values are defined in FC-LS-3.

6.3.16 Read Exchange Concise

The Read Exchange Concise (REC) ELS is used by a channel that supports transport-mode operations to determine the state of a TCW I/O operation at a control unit. General aspects of the REC ELS, including the format of the REC ELS command and response, are given in FC-LS-3.

The REC request payload identifies the OX_ID and RX_ID of the transport-mode operation for which status is to be provided. If the control unit receiving the REC ELS determines that the OX_ID or RX_ID in the request payload are inconsistent, then it shall reply with an LS_RJT Sequence with a reason code of unable to perform command request and a reason code explanation of invalid OX_ID-RX_ID combination. The Parameter field in the frame header of an REC ELS and of an LS_ACC in response to an REC ELS is not used by FC-SB-6. See 9.5.5 for FC-SB-6 usage of the REC ELS.

When the reply to the REC ELS is LS_ACC, the FC4VALUE field in the LS_ACC payload should be set to zero and the E_STAT field in the payload shall be set as defined in FC-LS-3.

The FC4VALUE may be set to a non-zero value for read and write operations. For read and write operations, the FC4VALUE should be set to:

a) For a write operation, the number of bytes successfully received by the device. Data that has been retransmitted or overlaid shall be counted only once;

b) For a read operation, the number of bytes transmitted by the device. Data that has been retransmitted or overlaid shall be counted only once; or

c) For a bidirectional operation, 00000000h.

6.4 FC-SB-6 Link-Control Functions

6.4.1 FC-SB-6 Link-Control Function Overview

FC-SB-6 link-control functions provide the means by which the logical paths between a channel and control-unit are established and maintained. FC-SB-6 link-control functions also provide information about conditions on the physical and logical paths that affect the transmission or reception of information units (IUs). These functions are performed by means of control IUs containing a link-control DIB (link-control IUs - see 8.12). A link-control IU containing a link-control DIB with a link header specifying one of the link-control functions, is referred to by the name of the link-control function specified in the link header (e.g., when the link-control function is a request to establish a logical path, the link-control IU is referred to as an establish logical path IU - see table 4 for a list of the link-control functions).

FC-SB-6 link-control functions are performed primarily during initialization or when certain error conditions occur on a link. Link-control IUs shall only be sent in command mode.

FC-SB-6 link-control requests shall be part of a request-response pair, followed by a link-level acknowledgment (LACK) response. The link-control IU containing the request (link-control-request IU) shall be sent, opening an exchange, as an unsolicited control information category. The exchange containing the link-control-request IU shall be left open. The link-control IU containing the response (link-control-response IU) shall be sent as a solicited control information category in a new exchange. The exchange containing the link-control-response IU shall be closed. When the sender of the link-control-request IU receives the link-control-response IU, it shall send a link-level-acknowledgment IU as a solicited control information category on the same exchange used to send the link-control-request IU. The exchange containing the link-level-acknowledgment IU shall be closed. An example of an FC-SB-6 link-control function is a link-control-request IU sent by the channel to request the establishment of a logical path, a link-control-response IU sent by the control unit indicating whether or not the logical path was established, and a link-level-acknowledgment IU sent by the channel to close the outbound exchange.

Link-control-request IUs shall always be sent on a new exchange (i.e., they shall never be sent on an existing exchange). If a link-control-request IU is received on an existing exchange, an FC-SB-6 link-level protocol error shall be detected.

If a channel or control unit receives an IU when any of the following apply, then the IU shall be discarded and a link-level protocol error shall be detected:

a) The received IU is a link-control-response IU and no link-control-request IU was sent, or a link-control-request IU was sent and the link-control-response IU received is not an allowed response;

b) The received IU is a link-control IU specifying a link-control function that is not recognized;

c) The received IU has an information category other than an information category allowed for the IU; or

d) The received IU is a link-control IU with a link-payload byte-count field set to a value different from that described in table 4.

Table 4 summarizes the FC-SB-6 link-control functions and the contents of fields within the IU. For a definition of the format of fields in link-control IUs, see figure 16 and 8.12.

Table 4 – Summary of Link-Control Request and Response IUs

IU	Sent by Channel	Sent by Control Unit	Expected Reply or Reply to	FC-SB-6 Header		Link Header		Link Payload
				Channel Image ID	**Control-Unit Image ID**	**Link-Control Field**	**Link-Control Info. Field**	
Link-Control Request IUs								
ELP	yes	no	LPE,LPR	Channel Image ID	Control-Unit Image ID	0100 0001	Optional Features	None
RLP	yes	no	LPR	Channel Image ID	Control-Unit Image ID	0100 1001	n/a	None
TIN	yes	yes	TIR	Function Dependent^1	Function Dependent^1	0000 1001	Function Identifier^6	None
Link-Control Response IUs								
LPE	no	yes	ELP	Channel Image ID	Control-Unit Image ID	0101 0001	Optional Features	None
LPR	no	yes	ELP,RLP	Channel Image ID	Control-Unit Image ID	0101 1001	Reason	None
TIR	yes	yes	TIN	Function Dependent^2	Function Dependent^2	0000 0001	Function Identifier^7	Initialization/ Capability State^2
LRJ	yes	yes	Initiation IU^5	Note^3	Note^3	0001 0001	Reason Code	None
LBY	yes	yes	Initiation IU^5	Note^3	Note^3	0010 0001	n/a	None
LACK	yes	yes	Note^4	0	0	0110 0001	n/a	None

Notes:

1 See 6.4.7 for a description of the use of the channel image ID and control-unit image ID fields in the TIN IU.

2 See 6.4.8 for a description of the use of the channel image ID field, control-unit image ID field, and link payload in the TIR IU.

3 This field is set to the same value as the corresponding field in the discarded initiation IU.

4 See 6.4.6 for a description of the possible uses for the LACK IU.

5 A valid reply to all Initiation IUs except initiation IUs sent in transport mode

6 The TIN IU is used to perform both the TIN and TINC functions- when bit 7 of byte 0 of the link-control-information field is zero, the TIN function is performed (see 6.4.7.2); when the bit is set to one, the TINC function is performed (see 6.4.7.3).

7 The TIR IU is used to perform both the TIR and TINCR functions- when bit 7 of byte 0 of the link-control-information field is zero, the TIR function is performed (see 6.4.8.2); when the bit is set to one, the TINCR function is performed (see 6.4.8.3).

The functions performed by the FC-SB-6 link-control IUs in the above table are described in clause 6.

6.4.2 Establish Logical Path

The establish logical path (ELP) function shall be sent from a channel image to a control-unit image to indicate the optional features supported by the channel and to request the establishment of a logical path. The number of CTC connections for which the channel is providing emulated control unit functionality may also be indicated. A logical path, when established, shall identify the channel and control-unit images and shall specify the optional features that are used when IUs are sent between the channel and the control unit. Optional features are alternate methods of operation that may be used in place of the default method of operation.

A channel shall perform the ELP function by sending an ELP IU to the control unit. The link-control DIB shall contain a link header with a link-control field specifying the ELP function, a link-control-information field specifying optional features, and a CTC counter field that may specify the number of CTC connections for which the channel is providing emulated control unit functionality (see 8.12.2). The link-control DIB shall not contain a link payload. The normal response to an ELP IU is the logical path established IU.

The channel image for the path to be established shall be identified by the combination of the channel N_Port ID and channel image ID.

The control-unit image shall be identified by the combination of the control-unit N_Port ID and control-unit image ID.

All control units shall be able to accept any value in the channel image ID field. The control unit may still restrict the quantity of logical paths permitted based on resources and system requirements.

Although a control unit that does not allow multiple control-unit images may restrict the number of logical paths it accepts to one, that one logical path may have any value of channel logical address.

The optional features that the channel supports shall be specified in the link-control-information field (see 8.12.2.3).

Each bit in the link-control-information field corresponds to a different optional feature. The bit for an optional feature shall be set to one if that optional feature is supported; otherwise, the bit shall be set to zero. Bits for which no optional feature is defined shall be set to zero by the channel and ignored by the control unit.

Optional features for the following bits in the link-control information field have been defined:

Bit	*Meaning*
0	Enhanced CRC Generation: When bit zero of the link-control information field is set to zero in the ELP IU, optional CRC generation initialization is not supported. When bit zero of the link-control information field is set to one in the ELP IU, optional CRC generation initialization is supported. See 8.6.5 for additional information.
1	Persistent IU Pacing: When bit one of the link-control information field is set to zero in the ELP IU, persistent IU pacing is not supported. When bit one of the link control information field is set to one in the ELP IU, persistent IU pacing is supported. See 9.2.2.5 for additional information.
15	CTC Connection: When bit 15 of the link-control information field is set to zero in the ELP IU, the channel does not support control-unit emulation for a CTC connection on the indicated

logical path. When bit 15 of the link-control information field is set to one in the ELP IU, the channel does support control-unit emulation for a CTC connection on the indicated logical path. When control-unit emulation for a CTC connection is supported, the value specified in the CTC counter field is valid (see 8.12.2.4).

When a control unit which is not an emulated control unit receives the ELP IU, it shall compare the optional features of the channel with its own supported optional features. The optional features to be used shall be those features which both the channel and the control unit support.

When an emulated control unit receives an ELP IU, the optional features to be used are those optional features which both channels support with the exception of the CTC connection feature. A channel supporting CTC connection may use the CTC connection optional feature and respond with an LPE IU regardless of whether the other channel supports CTC connection. If the sending channel has indicated support for CTC connection by setting the CTC connection bit to one in the ELP IU, the CTC connection bit in the LPE IU is set to one; otherwise the CTC connection bit is set to zero.

The requested logical path shall be established provided that the optional features for the requested logical path do not affect the optional features associated with other previously established logical paths with the same channel. If the request to establish a logical path would affect the optional features associated with other previously established logical paths with the same channel, the requested logical path shall not be established, and an LPR IU shall be sent in response to the ELP IU with an appropriate reason code (see 6.4.5).

When a logical path is established, the optional features which both the channel and the control unit support shall be saved and associated with the logical paths for that channel in such a way that all subsequent IUs for which those features apply are transferred on that logical path using those optional features.

A channel shall attempt to establish logical paths to the control-unit images that are described in its configuration definition. This shall be done when a channel image is initialized, when configuration changes are made, or when the channel receives an indication that the logical path no longer exists. The generation of a configuration definition is beyond the scope of this standard.

Each logical path shall be established with a separate exchange of IUs, an ELP request IU and an LPE response IU. Failure to establish a logical path shall not affect other existing logical paths or the ability to establish other logical paths.

If an ELP IU is received with a request for a logical path that is already established, the IU shall be accepted, provided that no errors are detected. If the establishment of the logical path would not affect optional features associated with other previously established logical paths with the same channel, a system reset shall be performed with respect to that logical path and a logical path established response shall be sent. The logical path established response does not indicate initiation or any degree of progress made for the associated system reset.

The channel shall not consider the logical path to be established until it receives an error-free logical path established IU. The control unit shall not consider the logical path to be established until the necessary action is taken at the control unit to establish the logical path, and the sending of an error-free logical path established IU is completed. If an error is detected when an ELP IU is received, the appropriate response, if any, shall be made, and the logical path shall not be established. If the control-unit image is not able to perform the ELP function for reasons other than an error condition or a link-busy condition, a logical path removed IU containing the appropriate logical path removed reason code shall be sent in response. If a logical path removed IU is received in response to an ELP IU, the logical path shall be considered not established. If a logical path removed IU response is received, the

ELP IU may be retried until the logical path is established. The number of retries is model dependent. In the case of logical path removed, whether or not the ELP is retried shall depend on the reason code (see 6.4.5).

If a channel receives an ELP IU with a TYPE field of hex'1B', a link-level-reject (LRJ) IU with a reject-reason code of protocol error shall be sent in response if the conditions for sending an LRJ IU are satisfied. See 6.4.9 for additional information. Otherwise, when CTC connection is supported, a Logical Path Established (LPE) IU is sent to complete the establishment of the logical path.

In the event that there is an error in the response to a channel request to establish a logical path, the channel shall not assume that the requested action has or has not taken place. If no valid response is received by the channel to the ELP request, the channel may retry the request. The number of retries is model dependent.

6.4.3 Remove Logical Path

The remove logical path (RLP) function shall request the control unit to remove a logical path. A channel shall perform the RLP function by sending an RLP IU to the control unit. The link-control DIB shall contain a link header with a link-control field specifying the RLP function and no link payload.

The logical path to be removed shall be specified by the combination of the channel N_Port ID, the channel image ID, the control-unit N_Port ID, and the control-unit image ID.

An RLP IU shall be sent when a change in the channel-path configuration requires a control unit to be either physically or logically removed.

The removal of a logical path shall cause the control-unit image and its associated devices to be logically removed from the channel path. When an RLP IU is received and accepted, the logical path shall be removed, and the equivalent of a system reset shall be performed only for the affected logical path; that is, only the control-unit image associated with the logical path shall be affected, and only those operations and allegiances within the control-unit image for this logical path shall be reset (see 9.4.4 for information regarding system reset). The logical path removed IU shall be the normal response. Other logical paths associated with the same channel or different channels and the allegiances maintained to them shall be unaffected. After a logical path is removed, IUs for device-level functions shall not be sent or received using that logical path.

An RLP IU shall be received over the same physical path over which the logical path was established. If an RLP IU is received for a logical path that does not exist, the RLP shall be accepted, provided that no errors are detected, and the logical path removed response shall be sent.

If an error is detected when an RLP IU is received, the IU shall be discarded, the specified logical path is not removed, and the appropriate response, if any, for the error recognized shall be sent. The channel shall not consider the logical path removed until it receives the LPR IU and no errors are detected. The control unit shall not consider the logical path removed until the logical path removed IU is sent.

Unless the channel receives a valid response to an RLP, the channel shall not assume that the requested action has or has not taken place. If an invalid response is received by the channel to the remove-logical-path request, the channel shall retry the request. The number of retries is model dependent.

6.4.4 Logical Path Established

The logical path established (LPE) function shall confirm the successful completion of an ELP request and the establishment of the logical path, and shall indicate the optional features to be used on all IUs sent between the channel and the control unit. An optional feature shall be used only if the channel has indicated support for that optional feature, the control unit supports the optional feature, and use of the feature does not affect the optional features associated with other previously established logical paths with the same channel. A control unit shall perform the LPE function by sending an LPE IU to the channel. The link-control DIB shall contain a link header with a link-control field specifying the LPE function and a link-control-information field specifying optional features. The link-control DIB shall not contain a link payload.

The logical path which was established shall be identified by the combination of the control-unit N_Port ID, control-unit image ID, the channel N_Port ID, and the channel image ID in the LPE IU. The optional features to be used shall be specified in the link-control-information field. See 8.12.2.3.

Each bit in the link-control-information field shall correspond to a different optional feature. The bit for an optional feature shall be set to one if that optional feature is to be used; otherwise, the bit shall be set to zero. See 6.4.2 for information concerning the CTC connection optional feature. Bits for which no optional feature is defined shall be set to zero by the control unit and ignored by the channel.

See 6.4.2 for a definition of the bits in the link-control information field for which optional features have been defined.

When a channel receives an LPE response to the ELP IU, the indicator for the requested logical path shall be established provided that the optional features for the requested logical path do not affect the optional features associated with other previously established logical paths with the same control unit. When the logical path is established, the optional features to be used shall be saved and associated with the logical paths for that control unit in such a way that all subsequent IUs for which those features apply are transferred on that logical path using those optional features. If establishment of a logical path would affect the optional features associated with other previously established logical paths, an RLP IU shall be sent to remove the logical path.

When an ELP IU is accepted, the LPE IU shall be the normal response. A logical path shall be considered not established by the recipient of an ELP IU until it has sent the LPE IU and shall be considered not established by the sender of an ELP IU until the LPE IU is received.

6.4.5 Logical Path Removed

The logical path removed (LPR) function shall confirm the successful completion of an RLP request and the removal of a logical path. A control unit shall perform the LPR function by sending an LPR IU to the channel. The link-control DIB shall contain a link header with a link-control field specifying the LPR function and a link-control-information field specifying a reason code. The LPR IU shall have no link payload.

The combination of the control-unit N_Port ID, control-unit image ID, the channel N_Port ID, and the channel image ID in the LPR IU shall identify the logical path that was removed.

An LPR IU shall be the normal response to an RLP request. A logical path shall be considered not removed by the recipient of an RLP frame until it has sent the LPR IU in response and shall be considered not removed by the sender of the RLP IU until the LPR IU is received.

An LPR IU may also be sent in response to an ELP IU when no error or link-busy conditions are detected but the requested logical path is not established.

The first byte of the link-control-information field in the link header is defined as follows:

Bit	*Meaning*
0-3	Bits 0-3 of the link-control-information field in the link header shall be reserved for future use; these bits shall be set to zeros by the sender of the LPR IU and shall be ignored by the recipient of the IU.
4-7	Bits 4-7 of the link-control-information field shall contain a reason code which indicates why the logical path was removed. The values (in binary) for the following reason codes shall have these meanings:

Value	*Meaning*
0000	This LPR IU is a response to RLP.
0001	The supported optional features received would affect those in use on already established logical paths with the same channel. The ELP should not be retried for this condition.
0010	The control-unit image has no resources available for establishing new logical paths on this physical path. The channel may retry the ELP a model-dependent number of times.
0011	Device-level initialization is not complete; the control unit is not ready to perform device-level operations. The channel should retry the ELP a model-dependent number of times.
0100	A control-unit image corresponding to the control-unit image ID field in the FC-SB-6 header of the ELP IU does not exist. The channel should not retry the ELP for this condition.
0101- 1111	Reserved.

A link-level protocol error shall be detected if an LPR IU containing a reserved reason-code value is received.

6.4.6 Link-Level Acknowledgment

The LACK function shall be used to close an exchange used to initiate a link-control request, and exchanges which can not be closed with either a link-control response or a device-level-control function. Examples of exchanges which may be ended with the LACK IU are as follows:

a) An exchange left open as a result of a link-control request;

b) An exchange left open as a result of a cancel, system reset, or request status sent in an initiation IU as information category unsolicited control which received a DACK IU as an allowed response;

c) An exchange left open as a result of a selective reset device-level control function which received a DACK IU as an allowed response;

d) An exchange left open as a result of an initiation IU receiving an LRJ IU or LBY IU as an allowed response; or

e) An exchange left open as a result of a command or a device-level-control function which received an address exception IU as an allowed response.

The LACK function shall be performed by sending the LACK IU. The link-control DIB shall contain a link header with a link-control field specifying the LACK function and no link payload.

The channel and control-unit image IDs of the LACK IU shall be set to zero by the sender and ignored by the recipient.

6.4.7 Test Initialization

6.4.7.1 Test Initialization Overview

The test-initialization (TIN) IU provides a method for determining which logical paths are considered established for a channel image or control-unit image, and for determining which logical paths are capable of being established for a channel image with a control unit.

When the TIN IU is used to determine the logical paths that are considered established for a channel image or control-unit image, the function performed by the TIN IU is defined as the TIN function; when the TIN IU is used to determine the logical paths that are capable of being established for a channel, the function performed by the TIN IU is defined as the TIN Capability (TINC) function. The TIN function is performed when the TIN IU contains a link header with bit 7 of byte zero of the link-control-information field set to zero. The TINC function is performed when the TIN IU contains a link header with bit 7 of byte zero of the link-control-information field set to one.

The TIN and TINC functions are described in the following sections.

6.4.7.2 Test Initialization Function

The test-initialization (TIN) function shall provide a method for determining which logical paths are considered established for a channel image or control-unit image. The TIN function shall be performed by sending a TIN IU that specifies the TIN function and is specified in either basic or extended mode. Basic or extended mode shall be specified by the value of the image-ID count field contained in byte one of the link-control information field. If the value of the count field is zero, a basic-mode TIN function is specified. If the value is non-zero, an extended-mode TIN function is specified. The link-control DIB shall contain a link header with a link-control field specifying hex '09' with bit 7 of byte zero of the link-control-information field set to zero. No link payload is provided for the TIN function. A TIN IU that specifies the TIN function is referred to as the TIN function and a TIR IU that specifies the TIR function is referred to as the TIR function.

It shall be model dependent as to whether the channel or control unit uses basic or extended mode; however, if basic mode is used, the receiver shall correspondingly use basic mode for the TIR function in response; otherwise a link-level protocol error shall be recognized. Unless stated otherwise, text pertaining equally to both basic and extended modes will not be referenced by either mode. Whenever text pertains to a particular mode of operation, the appropriate mode will be specified.

A channel or control unit may send the TIN function at any time; however, when initiative to perform the test-initialization function is generated, the TIN function shall be sent. The occurrence of any of the following events at a channel or control unit shall create the initiative to send a TIN function as follows:

a) If an RSCN is accepted by the N_Port of a channel or a control unit and one or more of the affected N_Ports is the N_Port of a channel or control unit to which one or more logical paths are indicated as being established, initiative to send a TIN function to each channel or control unit to which a logical path is indicated as being established shall be generated;

b) If the N_Port of a channel receives the LOGO ELS request from the N_Port of a control unit at a time when internal indicators at the channel indicate that one or more logical paths exist with the control unit, initiative to send a TIN function to the control unit shall be generated;

c) If the N_Port of a control unit receives a PLOGI ELS request from the N_Port of a channel at a time when one or more logical paths are indicated as being established to the channel, initiative to send a TIN function to the channel shall be generated; or

d) If a control unit exits the FC-FS-4 link failure state at a time when one or more logical paths are indicated as being established to one or more channels, initiative to send a TIN function to each channel to which a logical path is indicated as being established shall be generated.

Once initiative to send a TIN function to either a channel or control unit is generated, subsequent occurrences of any of the above events shall not create initiative to send another TIN function to the same channel or control unit if initiative to send the first TIN function has not been discharged. For a channel, initiative to send a TIN function to a control unit shall be discharged after the TIN function has been sent and a TIR function has been received in response or an event occurs which requires the removal of all logical paths to the control unit. For a control unit, initiative to send a TIN function to a channel shall be discharged if one of the following occurs:

a) A TIN function has been sent to the channel and a TIR function is received in response;

b) A logical path time-out condition for the channel has been recognized by any control-unit image, and the control unit has attempted to send a TIN function to that channel after recognizing the time-out condition, (see 10.2.4); or

c) An event occurs which requires the removal of all existing logical paths to the channel.

Upon accepting a TIN function, the recipient shall check whether it has logical paths with the source of the TIN function. The normal response is a TIR function.

When a channel sends a TIN function in basic mode, the logical paths to be tested for establishment shall be specified by the channel N_Port ID and channel image ID. The control-unit image ID field shall be set to zero by the channel and ignored by the control unit.

When a control unit sends a TIN function in basic mode, the logical paths to be tested for establishment shall be specified by the control-unit N_Port ID and control-unit image ID. The channel image ID field shall be set to zero by the control unit and is ignored by the channel.

When the channel sends a TIN function in extended mode, the set of logical paths to be tested for establishment shall be specified by the channel N_Port ID, the channel image ID, and an 8-bit image-ID count value contained in byte one of the link-control information field. The range of logical paths to be tested shall be determined starting with the channel image ID and incrementing to the smaller of a) the extent specified by the 8-bit image-ID count value, or b) a maximum channel image ID of 255. The image-ID count field shall be interpreted as an 8-bit unsigned binary integer, and the value in the field may range from 1 to 255. The value in the image-ID count field indicates the number of channel images beyond 1 to be tested. (e.g., a channel image ID of 2 and a count of 3 will test the 4 channel images 2, 3, 4, and 5). If the value in the image-ID count field is greater than 253, it shall be regarded by the control unit as an image-ID count of 253. The control-unit image ID shall be set to zero by the channel and is ignored by the control unit.

When the control unit sends a TIN function in extended mode, the set of logical paths to be tested for establishment shall be specified by the control-unit N_Port ID, the control-unit image ID, and an 8-bit image-ID count value contained in byte one of the link-control information field. The range of logical paths to be tested for establishment shall be determined starting with the control-unit image ID and incrementing to the smaller of a) the extent specified by the 8-bit image-ID count value, or b) a maximum control-unit image ID of 255. The image-ID count field shall be interpreted as an 8-bit unsigned binary integer, and the value may range from 1 to 255. The value in the image-ID count field indicates the number of control-unit images beyond 1 to be tested. (e.g., a control-unit image ID of 2 and a count of 3 will test the 4 control-unit images 2, 3, 4, and 5). If the value in the image-ID count field is greater than 253, it shall be regarded by the channel as an image-ID count of 253. The channel image ID shall be set to zero by the control unit and ignored by the channel.

NOTE 16 – The maximum upper limit of the contents of the image-ID count field is specified as 253 to allow the payload of the extended TIR function to be contained within a single IU. If additional images beyond the specified maximum of 254 need to be verified, then additional basic-mode or extended-mode TIN IUs need to be sent that specify the TIN function.

6.4.7.3 Test Initialization Capability Function

The test-initialization capability (TINC) function provides a method for a channel to determine the logical paths that are capable of being established between a channel image and a control unit. The TINC function shall be performed by the channel sending a TIN IU that specifies the TINC function to a control unit. The link-control DIB shall contain a link header with the link-control field specifying hex ‘09’ with bit 7 of byte zero of the link-control-information field set to one. No information is provided in the link payload for the TINC function. A TIN IU that specifies the TINC function is referred to as the TINC function and a TIR IU that specifies the TINCR function is referred to as the TINCR function.

A channel may send a TIN IU that specifies the TINC function at any time. Upon receiving a TIN IU that specifies the TINC function, the control unit determines the logical paths that it is capable of establishing with the channel images specified in the TIN IU and performs the TINCR function. The set of logical paths the control unit considers capable of being established includes all logical paths that are already established and logical paths that are permitted to be established according to the control-unit configuration. Depending on the resources available at the control unit, it may not be possible to establish all the indicated logical paths at a given time.

When the channel sends a TIN IU that specifies the TINC function, the set of logical paths to be tested for establishment capability shall be specified by the channel N_Port ID, the channel image ID, and an 8-bit image-ID count value contained in byte one of the link-control information field. The range of logical paths to be tested shall be determined starting with the channel image ID and incrementing to the smaller of a) the number of images specified by the 8-bit image-ID count value, or b) a maximum channel image ID of 255. The image-ID count field shall be interpreted as an 8-bit unsigned binary integer, and the value in the field may range from 0 to 255. The value in the image-ID count field indicates the number of channel images to be tested. (e.g., a channel image ID of 2 and a count of 3 will test the channel images 2, 3, and 4). This differs from the definition of the image-ID count field in the TIN function. If the value in the image-ID count field is greater than 254, it shall be regarded by the control unit as an image-ID count of 254. If the value of the image-ID count is zero, it shall be regarded by the control unit as an image-ID count of 1. The control-unit image ID shall be set to zero by the channel and is ignored by the control unit.

NOTE 17 – The maximum upper limit of the contents of the image-ID count field is specified as 254 to allow the payload of the TINCR function to be contained within a single IU. If additional images beyond the specified maximum of 254 need to be verified, then additional TIN IUs that specify the TINC function need to be sent.

6.4.8 Test Initialization Result

6.4.8.1 Test Initialization Result Overview

The test-initialization-result (TIR) IU shall be used to confirm the successful completion of either the TIN function or the TINC function.

When the TIR IU is sent to confirm successful completion of the TIN function, the function performed by the TIR IU is defined as the TIR function. When the TIR IU is sent to confirm successful completion of the TINC function, the function performed by the TIR IU is defined as the TINCR function. The TIR function is indicated when the TIR IU contains a link header with bit 7 of byte zero of the link-control-information field set to zero. The TINCR function is indicated when the TIR IU contains a link header with bit 7 of byte zero of the link-control-information field set to one. A TIR IU shall be sent only as a response to a TIN IU that is accepted.

The TIR and TINCR functions are described in the following sections.

6.4.8.2 Test Initialization Result Function

The test-initialization-result (TIR) function shall confirm the successful completion of the TIN function and shall indicate which logical paths are considered established for the channel or control unit performing the TIR function. The TIR function shall be specified in either basic or extended mode as determined by the value of the image-ID count field contained in byte one of the link-control information field. If the value is zero, a basic-mode TIR function is specified. If the value is non-zero, an extended-mode TIR function is specified. The TIR IU for the TIR function shall contain a link header with a link-control field specifying hex '01' with bit 7 of byte zero of the link-control-information field set to zero.

The TIR function shall be performed by sending a TIR IU that specifies the TIR function. A basic-mode TIR function shall be sent in response to a basic-mode TIN function and may also be sent in response to an extended-mode TIN function. A basic-mode TIR function may be sent in response to a TINC function when the control unit does not support the FC-SB-6 TINC function. An extended-mode TIR function shall only be sent in response to an extended-mode TIN function; otherwise, a link-level protocol error shall be recognized.

When a TIR function is sent in basic mode, the link-control DIB shall contain a 32 byte logical path field in the link payload (see table 5 for the format of the logical path field).

If a channel sends the basic-mode TIR function, the channel image ID field shall be set to zero by the channel and ignored by the control unit. The control-unit image ID field shall be the same as the control-unit image ID field received in the TIN function. If a control unit sends the basic-mode TIR function, the channel image ID field in the SB-4 header shall be the same as the channel image ID field received in the TIN function, and the control-unit image ID field shall be set to zero by the control unit and ignored by the channel.

The 32 byte link payload for the basic-mode TIR function has the format given in table 5.

Table 5 – Logical Path Field - Basic

Word	Contents
0	Logical Paths 0 - 31
1	Logical Paths 32 - 63
2	Logical Paths 64 - 95
3	Logical Paths 96 - 127
4	Logical Paths 128 - 159
5	Logical Paths 160 - 191
6	Logical Paths 192 - 223
7	Logical Paths 224 - 255

The logical-path field shall indicate whether each of the 256 possible logical paths are considered established. There is a bit in the logical-path field for each possible logical path that may be established with the source of the TIN function. Each bit shall correspond to an image ID. Starting with bit 0, word 0, logical addresses 0 through 255 are assigned in ascending order. The bit for a logical address corresponding to a logical path shall be set to one if that logical path is established and shall be set to zero otherwise (see 3.3.3 for a definition of the SB-4 bit numbering convention used in table 5).

When the basic-mode TIR function is received in response to a TIN function and one or more logical paths are indicated as not established when they were previously considered to be established, a test-initialization-result error shall be recognized. If one or more logical paths are indicated as being established when they were previously considered not established, a test-initialization-result error shall be recognized. For recovery from a test initialization result error, see 11.2.9. If a test-initialization-result error is not recognized, no action shall be taken.

When an extended-mode TIR function is sent, the link-control DIB shall contain a link header with the link-control field specifying the TIR function, an 8-bit image-ID count value contained in byte one of the link-control information field, and a logical path field in the link payload. The contents of the link payload shall specify the logical paths that are recognized as being established by the receiver of the extended TIN function, and which correspond to the range of image IDs specified in the received extended TIN function. The image-ID count indicates the range of logical paths contained in the response starting with the control-unit image ID or channel image ID specified in the TIN being responded to and incrementing to the smaller of a) the extent specified by the 8-bit image-ID count value contained in byte one of the link-control information field of the TIN function; or b) a maximum control-unit image or channel image ID of 255. The image-ID count field shall be interpreted as an 8-bit unsigned binary integer and the value in the field may range from 1 to 253. The value in the image-ID count field indicates the number of control-unit images or channel images beyond 1 that were tested and for which results are being provided. (e.g., a count of 3 indicates 4 images were tested). The length of the payload specified by the link-payload byte count contained in the DIB header shall be 32 plus 32 times the value of the image-id count field, and may vary between 64 and 8128 bytes (see table 6 for an example and definition of the format of the logical path field).

If the channel sends an extended-mode TIR function, the channel image-ID field shall be set to zero by the channel and ignored by the control unit. The control-unit image ID shall be the same as the control-unit image ID received in the extended TIN function. If a control unit sends an extended-mode TIR function, the channel image ID shall be the same as the channel image ID received in the extended TIN function, and the control-unit image ID shall be set to zero by the control unit and ignored by the channel.

An example of the link payload for an extended TIR function which would be returned in response to a TIN function with an image ID field set to zero and an image-ID count field set to 253 is shown in table 6.

.

Table 6 – Logical Path Field - Extended

Word	Contents
0	Logical Paths 0 - 31
1	Logical Paths 32 - 63
2	Logical Paths 64 - 95
3	Logical Paths 96 - 127
4	Logical Paths 128 - 159
5	Logical Paths 160 - 191
6	Logical Paths 192 - 223
7	Logical Paths 224 - 255
8	Logical Paths 256 - 287
9	Logical Paths 288 - 319
. . .	. . ,
2031	Logical Paths 64,992 - 65,023

In the example of table 7, the logical-path field indicates whether each of the 65,024 logical paths are considered established.

The first 256 bits of the logical path field shall relate to the set of logical paths corresponding to the first source image ID, and continuing, logical paths 256-511 shall correspond to the second source image ID, and subsequent groups of 256 logical paths likewise shall correspond to consecutively increasing source image IDs. The bit for a logical address corresponding to a logical path shall be set to one if that logical path is established and shall be set to zero otherwise.

6.4.8.3 Test Initialization Capability Result Function

The test-initialization-capability result (TINCR) function shall confirm the successful completion of the TINC function and shall indicate the logical paths that the control unit performing the TINCR function is capable of establishing with the channel that sent the TINC function and shall include logical paths that are already established.

Due to resource constraints, a control unit may only be able to establish a subset of the logical paths that it indicates it is capable of establishing at any one time.

The TINCR function shall be performed by sending a TIR IU that specifies the TINCR function. A TINCR function shall be sent only as a response to a TINC function.

When a TINCR function is sent, the link-control DIB shall contain a link header with the link-control field specifying hex ‘01’ and bit 7 of byte zero of the link-control-information field set to one, a logical path field in the link payload, and an 8-bit image-ID count value contained in byte one of the link-control information field. The image-ID count indicates the range of logical paths contained in the response starting with the channel image ID specified in the TINC being responded to and incrementing to the smaller of a) the number of images specified by the 8-bit image-ID count value contained in byte one of the link-control information field of the TINC function, or b) a maximum channel image ID of 255. The image-ID count field shall be interpreted as an 8-bit unsigned binary integer and the value in the field may range from 1 to 254. If the image-ID count value is zero or greater than 254, a link-level protocol error shall be recognized.

The contents of the link payload shall specify the logical paths that are recognized as being capable of being established by the control that received the TINC function and which correspond to the range of channel image IDs specified in the received TINC function. The length of the payload specified by the link-payload byte count contained in the DIB header shall be 32 times the value of the image-id count field, and may vary between 32 and 8128 bytes (see table 7 for an example and definition of the format of the logical path field).

The channel image ID in the TINCR shall be the same as the channel image ID received in the TINC function and the control-unit image ID shall be set to zero by the control unit and ignored by the channel.

An example of the link payload for a TINCR function which would be returned in response to a TINC function with an image ID field set to zero and an image-ID count field set to 254, is shown in table 7.

.

Table 7 – TINCR Logical Path Field

Word	Contents
0	Logical Paths 0 - 31
1	Logical Paths 32 - 63
2	Logical Paths 64 - 95
3	Logical Paths 96 - 127
4	Logical Paths 128 - 159
5	Logical Paths 160 - 191
6	Logical Paths 192 - 223
7	Logical Paths 224 - 255
8	Logical Paths 256 - 287
9	Logical Paths 288 - 319
. . .	. . ,
2031	Logical Paths 64,992 - 65,023

In the example of table 7, the logical-path field indicates which of the 65,024 logical paths can be established with the control unit, and includes logical paths that are already established.

The first 256 bits of the logical path field shall relate to the set of logical paths corresponding to the first source image ID, and continuing, logical paths 256-511 shall correspond to the second source image ID, and subsequent groups of 256 logical paths likewise shall correspond to consecutively increasing source image IDs. The bit for a logical address corresponding to a logical path shall be set to one if that logical path is capable of being established or is already established with the channel that sent the TINC function; otherwise, the bit shall be set to zero.

6.4.9 Link-Level Reject

The link-level-reject (LRJ) function shall be performed when a command-mode initiation IU was received and discarded by a channel or control unit because of an error condition. The LRJ function shall be performed by sending an LRJ IU. For a transport-mode initiation IU, the control unit sends a transport-response IU with status indicating link-level reject (see table 28) when the IU was received and discarded because of a logical-path-not-established condition. No other link-level reject conditions are defined for transport mode.

When an LRJ IU is sent, the channel and control-unit image ID fields shall be set to the corresponding values in the IU for which the reject is being sent. If the IU for which the reject is being sent contains other than a link-control DIB and the AS bit is set to one, then the AS bit shall be set to one in the LRJ IU and the device address is provided. If the AS bit is set to zero in the IU being rejected, the AS bit shall be set to zero in the LRJ IU.

The link-control DIB shall contain a link header with a link-control field specifying the LRJ function and a link-control information field specifying the reject-reason code. The DIB shall not contain a link payload.

A channel or control unit shall send an LRJ IU in response to a command-mode initiation IU when a logical path not established error condition is detected and all of the following conditions are satisfied:

a) The IU contains no FC-FS-4 errors;

b) The IU contains no exchange errors;

c) The IU is either an unsolicited control IU, an unsolicited command IU, or an unsolicited data IU; and

d) No other condition exists or has been recognized that requires the sending of either a link-level-busy IU or an FC-FS-4 primitive sequence.

An LRJ IU shall also be sent by a channel as the response to an ELP IU received with a TYPE field of hex'1B', provided all of the above conditions are satisfied and any of the following conditions are satisfied:

a) The channel does not support CTC connection or does not support CTC connection for the requested logical path (see 6.4.2);

b) The channel is not waiting for a response from an ELP IU previously sent to the channel from which it received the ELP IU, and the CTC counter field of the received ELP IU contains a valid CTC counter value which is less than the number of CTC connections for which the channel is providing the emulated control unit functionality (see 8.12.2.4);

c) The channel is not waiting for a response from an ELP IU previously sent to the channel from which it received the ELP IU, the CTC counter field of the received ELP IU contains a valid CTC counter value which is equal to the number of CTC connections for which the channel is providing the emulated control unit functionality, and the N_Port ID of the channel is less than the N_Port ID of the channel which sent the received ELP IU;

d) The channel is waiting for a response from an ELP IU previously sent to the channel from which it received the ELP IU, and the CTC counter field of the received ELP IU contains a valid value which is less than the CTC counter value specified in the ELP IU previously sent; or

e) The channel is waiting for a response from an ELP IU previously sent to the channel from which it received the ELP IU, the CTC counter field of the received ELP IU contains a valid value which is equal to the CTC counter value specified in the ELP IU previously sent, and the N_Port ID of the channel is less than the N_Port ID of the channel which sent the received ELP IU.

When an LRJ IU is received in response to an IU which initiates a connection, the type of error shall be indicated by the reject-reason code. If the LRJ IU is received in response to an IU sent during a connection, a link-level protocol error shall be recognized.

The LRJ IU shall have a two byte link-control information field. Bits 1-7 of the first byte of the link-control information field shall contain the reject-reason code. The link-control information field of an LRJ IU shall have the format shown in figure 15.

r	Reject-Reason Code	Reserved
0	1 7	8 15

Figure 15 – Link-Control Information Field of the LRJ IU

The reject-reason code shall appear in bits 1-7 of the link-control information field of an LRJ IU. This field shall identify the error condition that caused the IU to be rejected.

When a channel sends the LRJ IU, the only valid reject-reason codes shall be '000 0110'b, corresponding to the logical path not established reject-reason code, and '000 1001'b, corresponding to protocol error reject-reason code, respectively. All other reject-reason codes shall be reserved. When a control unit sends the LRJ IU, the only valid reject-reason code is '000 0110'b, corresponding to the logical path not established reject-reason code. All other reject-reason codes shall be reserved.

The protocol error reject-reason code is used by the channel to indicate an error condition if an ELP IU with a type code of hex'1B' is received. The logical path not established reject-reason code is used by the channel and the control unit when a logical path not established error condition is detected.

Bit zero and bits 8-15 of the link-control-information field shall be reserved for future use; these bits shall be set to zero by the sender and are ignored by the recipient of an LRJ IU.

A link-level protocol error shall be detected if an LRJ IU is received with a reserved reject-reason-code value.

6.4.10 Link-Level Busy

The link-level-busy (LBY) function shall be performed to indicate that a command-mode initiation IU was received and discarded by a channel or control unit because of a link-busy condition. The LBY function shall be performed by sending an LBY IU to the source of the IU which was discarded. The link-control DIB shall contain a link header with a link-control field specifying the LBY function and no link payload. The link-level busy function is not supported in transport mode.

The channel and control-unit image ID fields shall be set to the corresponding values in the IU for which the LBY IU is being sent. If the IU for which the LBY IU is being sent contains other than a link-control DIB and the AS bit is set to one, then the AS bit shall be set to one in the LBY IU and the device address shall be provided. If the AS bit is set to zero in the IU for which the LBY is being sent, the AS bit shall be set to zero in the LBY IU.

An LBY IU shall be sent as a response when an IU is not accepted because of a link-busy condition. Link-busy conditions for which the LBY IU is sent shall be temporary conditions which cause an initiation IU to be discarded such as when a self-initiated function is being performed or when an FC-SB-6 resource required for the reception of the initiation IU is temporarily unavailable.

An LBY IU shall be sent in response to a command-mode initiation IU only if all of the following conditions are satisfied:

a) The IU contains no FC-FS-4 error;

b) The IU contains no exchange errors;

c) The IU is either an unsolicited control IU, an unsolicited command IU, or an unsolicited data IU; and

d) No other condition exists or has been recognized that requires the sending of either a link-level-reject IU or an FC-FS-4 primitive sequence.

When an LBY IU is received in response to an initiation IU a temporary busy condition shall be recognized, and the link-level or device-level function may be immediately retried or retried at a later time. When an LBY IU is received in response to an IU while a connection exists, a link-level protocol error shall be detected.

7 N_Port Link Initialization

7.1 N_Port Link Initialization Overview

This clause describes the initialization procedures for a channel and control unit unique to this standard.

The initialization process establishes the necessary conditions for elements of a channel path to be able to sustain both link-level and device-level communication. Under normal conditions, the initialization process occurs infrequently (e.g., after a power on procedure or during a system-initialization procedure) and is not part of the execution of an I/O operation.

The initialization process may be thought of as a series of hierarchical steps.

For the channel, the steps of the initialization process and their associated procedures are:

1) link-initialization procedure;

2) channel login and security attribute determination procedure;

3) channel node-identifier-acquisition procedure;

4) channel state-change-registration procedure;

5) channel-link-incident-record-registration procedure;

6) channel process login procedure when FC-SB-6 process login is supported by the channel and control unit; and

7) channel logical-path-establishment procedure.

For the control unit, the steps of the initialization process and their associated procedures are:

1) link-initialization procedure:

2) control unit login procedure;

3) control unit node-identifier-acquisition procedure;

4) control unit state-change-registration procedure;

5) control unit process login procedure when FC-SB-6 process login is supported by the channel and control unit; and

6) control unit logical-path-establishment procedure.

If a procedure encounters an FC-LS-3 or FC-SB-6 response indicating a busy condition, the procedure shall be retried until either the busy condition no longer exists and the procedure is successful or a condition other than a busy condition is encountered, in which case, the protocols defined for that condition determine the action to be taken. The retry for a busy condition may be deferred until attempts to complete the procedure with other N_Ports have been made.

If a procedure is not successful because of an unrecoverable error, that portion of the initialization process shall be terminated and all or a portion of the channel or control unit shall be considered uninitialized, depending on which initialization procedures were not successful.

The hierarchy of initialization steps shall also be followed when there is an error or another event that causes regression within the initialization process. When an error or other event indicates that the results achieved at a previously completed step are no longer valid, information associated with that step and possibly subsequent steps in the hierarchy shall be discarded or may be considered no longer current for node-identifier information, and the initialization process for the affected channel or control unit shall be repeated for all affected steps (see 11.1 and 6.3.9). A node-identifier that is not current is an identification of the last known node attached to the N_Port, but as the result of some event, such as an FC-FS-4 link-failure condition, may not reflect the current attached node, since a configuration change may have occurred. For example, if a channel observes that a link which was previously considered operational is now not operational because of an FC-LS-3 link-failure condition, then the channel N_Port is implicitly logged out with the N_Port or F_Port at the other end of the link, and if the FC-FS-4 link failure condition persists for longer than SB_TOV, it removes any logical paths established for that link, and may discard the neighboring-node identifier or may consider the identifier to be not current. For this situation, the initialization process would start again with the link initialization procedure when the FC-FS-4 link failure condition no longer exists.

A Registered State Change Notification (RSCN) ELS command received from a Fabric Controller or N_Port which contains the N_Port ID of a channel or control unit for which the recipient of the RSCN considers the initialization process complete, may cause a test-initialization IU or RNID ELS command to be sent in order to verify the effect of the state change on previously established logical paths. If the test-initialization-result IU confirms that logical paths previously established by the initialization process are no longer considered established, the procedures shall be attempted, beginning at the appropriate step within the initialization-process hierarchy, to reestablish the logical paths.

An RSCN ELS command received from a Fabric Controller or N_Port which contains the N_Port ID of a channel or control unit for which the recipient of the RSCN considers the initialization process incomplete, may cause a test-initialization IU or RNID ELS command to be sent in order to verify the effect of the state change on logical paths previously established. If the test-initialization-response IU confirms that logical paths previously established by the initialization process are no longer considered established, the procedures shall be attempted, beginning at the appropriate step within the initialization-process hierarchy, to reestablish logical paths. If the test-initialization-result IU confirms that logical paths previously established are still valid, then the previously incomplete step shall be retried, and, if successful, the next step in the hierarchy shall be attempted for each logical path that is required on the channel path but that is not yet established. If the retry does not result in the successful completion of the initialization process, the recipient of the RSCN shall continue to consider the initialization process incomplete for that N_Port.

When an error or other event causes regression to the node-identifier acquisition step or to a previous step within the initialization process and the existing node-identifier of its neighbor is valid, the N_Port shall set the flag-field node-ID-validity code to the value 1 (node-ID is valid but may no longer be current) in the node descriptor for the affected neighboring node.

NOTE 18 – Many of the processes used during initialization require the appropriate action or response be taken within a prescribed period of time. The values selected for these periods of time are significantly greater than the worst-case propagation delay. In order that these time values not be compromised, it is important that each component in the path be designed to contribute the minimum amount of delay and that no one component in the path use a significant portion of the allowed time value. It should also be recognized that factors other than those directly associated with the procedure being performed may subtract from the time allowed. For example, when propagating a frame through a Fabric, the time required to route the frame through the Fabric should be taken into account.

7.2 Link-Initialization Procedure

Link initialization is described in FC-FS-4. When link initialization is complete, the N_Port or F_Port is in the active state.

Once link initialization is complete for an N_Port or F_Port, the N_Port or F_Port is considered to be operational as long as it remains in the active state (see FC-FS-4).

When an N_Port or F_Port exits the active state, the attached link is considered not operational, and the appropriate FC-FS-4 link initialization procedure shall be successfully completed in order to consider it operational.

7.3 Initialization Process for a Channel

7.3.1 Channel Initialization Overview

For a channel, the initialization process consists of the following procedures: link initialization, channel login and security attribute determination, channel node-identifier acquisition, channel state-change registration, channel link-incident-record registration, process login when supported by the channel and control unit, and channel logical-path establishment.

7.3.2 Channel Login and Security Attribute Determination

7.3.2.1 Overview

The N_Port of a channel shall attempt to perform channel login and security attribute determination following the completion of link-initialization. Channel login and security attribute determination consists of three steps: Fabric login, channel security attribute determination, and N_Port login. The login protocol is described in FC-LS-3 clause 6.

7.3.2.2 Channel F_Port Login

The N_Port of a channel, including the N_Port of a channel providing CTC connection, shall initiate channel login by sending an FLOGI ELS command with S_ID=hex'00 00 00' and D_ID=hex'FF FF FE'. The command shall be sent using Class 2 service, and support for Class 2 and Class 3 shall be indicated in the service parameters. If the channel is attached to a Fabric, the response from the Fabric assigns the channel an N_Port ID (Annex A, "Fabric Address Assignment", describes an address assignment scheme which allows the use of simplified configuration record formats). If an accept response is received, and if support for sequential delivery and Classes 2 and 3 service is indicated, then the channel shall proceed with channel security attribute determination. If errors prevent the successful receipt of an accept response, or if the service parameters of the response do not indicate support for sequential delivery and Classes 2 and 3 service, further steps in the initialization procedure shall not be performed.

If the N_Port of a channel receives an FLOGI ELS command before it is able to send an FLOGI ELS command, or if the response to the FLOGI ELS command indicates that the channel is not attached to a Fabric, then the N_Port of the channel shall proceed with N_Port login instead of F_Port login or Channel Security Attribute Determination.

7.3.2.3 Channel Security Attribute Determination

After the N_Port of a channel (including a channel providing CTC connection) has completed Fabric login in a multi-switch Fabric configuration, the N_Port of a channel shall perform channel security attribute determination. In single-switch Fabric configurations, channel security attribute determination is not required, and further steps of the initialization procedure are performed regardless of whether channel security attribute determination was performed (see 6.3.8 for additional information regarding the QSA ELS and Fabric configurations).

Channel security attribute determination is initiated by sending the QSA ELS command to the Fabric Controller. Change notification shall be enabled for Fabric binding and insistent domain ID. If the N_Port of a channel receives an accept response from the Fabric Controller, and if support for Fabric binding and insistent domain IDs is indicated, then the channel shall consider channel security attribute determination successful, and further steps in the initialization procedure shall be performed.

If errors prevent successful completion of the QSA ELS or if support for Fabric binding and insistent domain IDs is not indicated, then the channel shall consider channel security attribute determination unsuccessful. If channel security attribute determination is unsuccessful, further steps in the initialization process shall not be performed, the internal indicators for the logical paths previously considered established over the affected link shall be reset, and the associated device-level error recovery shall be performed (see 11.3.2.2).

7.3.2.4 Channel N_Port Login

7.3.2.4.1 Channel N_Port Login Overview

After the N_Port of a channel (including a channel providing CTC connection) has completed Fabric login and security attribute determination, or if the attempted Fabric login has revealed that the N_Port of the channel is directly attached to another N_Port, the N_Port of the channel shall perform N_Port login. When performing N_Port login, Class 2 service shall be used, and support for the following features shall be indicated:

a) Classes 2 and 3; and

b) continuously increasing relative offset for solicited data information category.

7.3.2.4.2 Channel N_Port Login: Point-to-Point Configurations

If the N_Port of the channel is directly attached to another N_Port, the N_Port of the channel shall perform login only with the attached N_Port. If no errors occur during N_Port login, and if support for Class 2, Class 3, and continuously increasing relative offset for solicited data information category is indicated, then the channel considers N_Port login successful, and further steps in the initialization process shall be performed. If errors occur, if support for Class 2 and Class 3 is not indicated, or if support for continuously increasing relative offset for solicited data information category is not indicated, then the channel considers N_Port login unsuccessful, and further steps in the initialization process shall not be performed.

7.3.2.4.3 Channel N_Port Login: Fabric Configurations

If the N_Port of a channel is attached to a Fabric, then after F_Port login and security attribute determination are complete, the N_Port of the channel shall initiate N_Port login with the N_Ports of each control unit which is configured to the channel, with each channel which supports CTC connection which is configured to the channel, with the management server, and with the Fabric Controller as required in further steps in the initialization process. When an initialization procedure requires communication with an N_Port with which the N_Port of the channel is not logged in, then the N_Port of the channel shall initiate N_Port login with that N_Port before performing the procedure. The channel also performs N_Port login with a control unit if it receives a PLOGI ELS from a control unit.

If no errors occur during N_Port login, if support for Class 2 and Class 3 is indicated, and (for the case of login with a control unit or with a channel which supports CTC connection) if support for continuously increasing relative offset for solicited data information category is indicated, then the channel shall consider N_Port login successful with respect to the N_Port, and further steps in the initialization process with respect to that N_Port shall be performed; otherwise, the channel shall consider N_Port login with that N_Port unsuccessful, and further steps in the initialization process with respect to an N_Port shall not be performed.

7.3.3 Channel Node-Identifier Acquisition

The channel shall attempt to acquire the neighboring-node identifier and, when the channel supports transport-mode operation, to acquire the node identifier of the N_Port for each control unit which is configured to the channel. The neighboring-node identifier refers to the identification of the node directly attached to the channel.

Channel node-identifier acquisition may be performed prior to channel N_Port login.

7.3.3.1 Channel Neighboring Node ID Acquisition

If the N_Port of a channel, including a channel providing CTC connection, is attached to another N_Port, the N_Port of the channel shall attempt to acquire the neighboring node identifier by sending an RNID ELS request to the attached N_Port after it has completed login with that N_Port. If the N_Port of the channel, including a channel providing CTC connection, is attached to a Fabric, the N_Port of the channel shall attempt to acquire the neighboring node identifier by sending an RNID ELS command to the Fabric Controller after it has completed login and security attribute determination.

If, an LS_RJT is received in response to the RNID ELS request to the neighboring node, or if errors occur which prevent the N_Port of the channel from sending the RNID ELS request or receiving the requested node-identifier, the channel shall consider node-identifier acquisition of its neighbor unsuccessful (see FC-LS-3 for information on LS_RJT).

When the channel is attached to a Fabric, the channel shall continue the initialization process even if the attempt to acquire the node-identifier of its neighboring node was unsuccessful. The lack of success in acquiring a valid and current neighboring-node identifier shall not prevent the establishment of logical paths or the execution of link-level and device-level functions. See 7.3.3.2 for handling of node-id acquisition failure in a point-to-point configuration.

The retry attempt to acquire the neighboring-node-identifier may be deferred for a link error, or an FC-LS-3 busy or reject indication. When the neighboring-node identifier is received, a channel that checks the node-descriptor flags field shall perform the following actions.

a) When the node-ID-validity code is zero, the channel shall establish the received 32-byte node descriptor as the node descriptor of its neighbor, and channel node-identifier acquisition is complete.

b) When the node-ID-validity code is not zero, the channel shall either:

    1) Check the existing node-descriptor node-ID-validity code if it had previously established the node descriptor of its neighbor and, if valid and not current, maintain the existing 32-byte node descriptor with an indication that the node-identifier is not current, or maintain an indication that the node descriptor of its neighbor is not valid, or

    2) Not check the existing node-descriptor node-ID-validity code and maintain an indication that the node descriptor of its neighbor is not valid.

When the node-ID-validity code is not zero, the channel shall retry channel node-identifier acquisition until a valid node descriptor is acquired. If there is reason to suspect that subsequent retries may not be successful, the node-identification procedure shall be suspended. If the channel has to provide node descriptors in a function such as link-incident reporting prior to acquisition of a valid and current node descriptor, the channel shall send the node descriptor of its neighbor with a node-ID-validity code of 1 or 2, as appropriate.

NOTE 19 – When a channel checks the neighbor-node-descriptor flag field and recognizes that the node-ID-validity code is not zero, the preferred implementation is to check the existing node-descriptor node-ID-validity code if it had previously established the node descriptor of its neighbor and, if valid and not current, maintain the existing 32-byte node descriptor with an indication that the node-identifier is not current, or maintain an indication that the node descriptor of its neighbor is not valid.

NOTE 20 – When a condition occurs that affects the validity of the neighboring-node identifier, the channel should consider the neighboring-node identifier as not current. This requirement applies for all conditions except for powering off or initial machine loading (IML), in which case, the channel may discard the neighboring-node identifier.

7.3.3.2 Channel Control Unit Node Id Acquisition

Channel control unit node ID acquisition shall be performed when the channel supports transport mode. When the channel is attached to a Fabric, the N_Port of the channel shall attempt to acquire the node identifier of each control unit N_Port which is configured to the channel by sending an RNID ELS request to each of those control unit N_Ports. In the case of a point-to-point configuration, the RNID ELS request is performed as part of the acquire-neighboring-node-ID procedure.

The actions taken by the channel when issuing the RNID ELS to a control unit are as follows:

a) If an error occurs which prevents the channel from sending the RNID ELS request, or if the RNID ELS is sent but no response is received or an LS_RJT is received, the channel shall consider node-identifier acquisition of the control unit unsuccessful. Initiative to issue the RNID ELS to the control unit may be periodically re-established and is re-established when a condition occurs such as an RSCN for the control unit that indicates conditions at the control unit may have changed. The lack of success in acquiring a valid and current control unit node identifier shall not prevent the establishment of logical paths or the execution of link-level and device-level functions.

b) If the RNID ELS request to a control unit N_Port completes with an LS_ACC and both the following conditions are true:

    1) the node-ID-validity code is zero in the node descriptor provided in the LS_ACC; and

    2) the node descriptor provided in the LS_ACC indicates support for FC-SB-6 process login,

    then the channel performs an FC-SB-6 process login prior to establishing logical paths to the control unit (see 7.3.6).

c) In all other cases, the channel does not perform process login with the control unit and attempts to establish logical paths to the control unit.

7.3.4 Channel State-Change Registration

The channel state-change-registration procedure enables the N_Port of a control unit or channel supporting CTC connection (in a point-to-point configuration), or the Fabric Controller (in a Fabric configuration) to send registered-state-change notifications to the N_Port of a channel (see 6.3.6).

If the N_Port of a channel, including a channel supporting CTC connection, is directly attached to another N_Port, the N_Port of the channel shall register for state-change notification by sending the SCR ELS command to the attached N_Port; however, if the channel is already registered to receive state-change notifications, it need not re-register. If the channel, including a channel supporting CTC connection, is attached to a Fabric, the N_Port of the channel shall register for state change notification by sending an SCR ELS command to the Fabric Controller; however, if the channel is already registered to receive state-change notifications, it need not re-register. For additional information on the SCR ELS command, see 6.3.7. After accepting the SCR ELS command, the Fabric Controller sends an RSCN ELS command to the N_Port of the channel when other N_Ports have potentially changed their states. When an accept response to the SCR ELS command is received, all IUs containing link-level information which are allowed for a channel may be sent unless link-incident-record registration is required. If errors occur which prevent the channel from sending the SCR ELS command or receiving an accept response, then the channel shall not proceed with further steps in the initialization process.

7.3.5 Channel Link-Incident-Record Registration

The channel link-incident-record-registration procedure enables control units which are not emulated control units and the management server to send RLIR ELS commands to the channel (see 6.3.10 and 6.3.11). The channel is not required to perform link-incident record registration with other channels providing CTC connection.

The N_Port of a channel shall perform the link-incident-record-registration procedure by sending the LIRR ELS command after it has completed the channel state-change registration procedure and before sending any other ELS command or FC-SB-6 IU. If the N_Port of a channel is directly attached to a control unit which is not an emulated control unit, the LIRR ELS command shall be sent to the N_Port of the control unit. If the N_Port of the channel is attached to a Fabric, the LIRR ELS command shall be sent to the management server. In addition for a Fabric configuration, link-incident-record registration shall be performed with each control unit with which the channel completes the N_Port login procedure; however, link-incident record registration is not required to be performed with other channels providing CTC connection.

If the N_Port of a channel receives an accept response to the LIRR ELS command, the channel shall consider link-incident-record registration successful with respect to the N_Port to which the request was sent, and the channel shall continue with further steps in the initialization process with respect to that N_Port. If an LS_RJT is received in response to the LIRR command, or if errors occur which prevent the N_Port of a channel from sending the LIRR ELS command to an N_Port or receiving an accept response, the channel shall consider link-incident-record registration unsuccessful with respect to the N_Port, and the channel shall continue the initialization process even though link-incident-record registration with the N_Port was unsuccessful. When the channel considers link-incident record registration unsuccessful, it is allowed to register for another type of link incident record, or for common link-incident records.

7.3.6 Process Login

When the channel supports transport-mode operation, the N_Port of the channel shall attempt to perform an FC-SB-6 process login with each control unit N_Port that is configured to the channel that supports FC-SB-6 process login as indicated by the node descriptor provided by the control unit (see 6.3.14).

The channel shall initiate the FC-SB-6 process login by sending a PRLI ELS request that specifies the FC-SB-6 type code to the control unit. If an LS_ACC response is received for the PRLI, logical path establishment (see 7.3.7) shall be performed with the control unit; otherwise logical path establishment shall not be performed with the control unit.

FC-SB-6 process login allows the channel and control unit to exchange various operating parameters (see 6.3.14), including whether the channel and control unit support transport-mode operation.

7.3.7 Channel Logical-Path Establishment

The last initialization procedure performed at the channel is the establishment of logical paths between the channel images sharing the N_Port and the control-unit images configured to the channel images. The channel logical-path-establishment procedure creates, at the channel, the information necessary for a particular channel image to communicate with a particular control-unit image to perform I/O operations.

The channel shall initiate the establishment of the logical path by sending an ELP IU. If the control unit is able to form a logical path between the channel image and control-unit image, it shall respond with the LPE IU indicating that the request is accepted and the logical path has been established; otherwise, it shall indicate that the logical path is not established and the reason why. When the logical path is established, the channel shall then allow device-level communication on that logical path between the specified channel image and control-unit image (see 6.4.2).

The channel shall consider the logical path to be established upon receiving an error-free LPE IU from the control unit. When the channel considers the logical path to be established, the channel shall consider the initialization process to be complete for the combination of that channel image and control-unit image. When the logical path is established, the N_Ports at each end of the physical path, and the device-level facilities associated with those N_Ports shall be considered operational and have the capability of performing their respective functions (see clause 8 for information about IUs containing device-level information).

A logical path shall no longer be usable for the exchange of IUs when either end of the path no longer considers the logical path to be established. A channel shall consider a logical path to be no longer established when either of the following events occurs:

a) The channel sends a RLP request to the control unit, and an error-free LPR response is returned (see 6.4.3); or

b) A channel detects an error for which the recovery action includes removing the logical path (see 11.3.2).

When a logical path is not established, the channel shall not perform device-level functions and protocols with respect to the corresponding control-unit image.

When the channel is initialized, one or more attempts to establish a logical path with each control-unit image in the channel-path configuration shall be made. The number of times beyond one that the channel attempts to establish the logical path for a control-unit image during initialization is model dependent.

A logical path may be removed by a control unit because of an FC-SB-6 link failure, a logical path timeout error, an FC-SB-6 offline condition, or a condition internal to the control unit; when this occurs, it may be asynchronous to the activity of the channel (see 11.3.2). If the control unit is not actively communicating with the channel image, the channel may not be immediately aware of the loss of the logical path. When the channel image later attempts to perform a device-level function using the logical path, abnormal conditions which affect the initialization of the channel are encountered (see 11.2.8).

During the initiation of a device-level function for a device or control unit, if the channel determines that a logical path, which it considers to exist, is to be removed, then the channel shall consider the device or control unit not operational with respect to the logical path and shall terminate the device-level function on this logical path. The channel shall also recognize a not-operational condition for all other devices for which an I/O operation was active or disconnected with respect to that logical path. All other devices associated with the logical path that is removed shall not be affected.

During the initiation of a device-level function for a device or control unit, if the channel determines that the logical path was previously not established, the channel shall attempt, by means of the initialization process, to establish the logical path before performing the device-level function. If the logical path is successfully established, the channel shall proceed to attempt to initiate the device-level function if the device-level function is still pending. If the initialization process is terminated during the initiation of the device-level function, the logical path shall remain not established and the channel shall consider the device or control unit to be not operational with respect to that logical path.

7.4 Initialization Process for a Control Unit

7.4.1 Control Unit Initialization Overview

For a control unit, the initialization process consists of the following procedures: link initialization, control unit login, control unit node-identifier acquisition, control unit state-change registration, process login, if supported, and control unit logical-path establishment.

7.4.2 Control-Unit Login

The N_Port of a control unit shall attempt to perform control-unit login following the completion of link-initialization. Control-unit login consists of two steps: F_Port login and N_Port login. For a description of F_Port and N_Port login protocol, see FC-LS-3.

7.4.2.1 Control Unit F_Port Login

Except in the case of an emulated control unit in which F_Port login is performed by the channel, the N_Port of a control unit shall initiate control-unit login by sending an FLOGI ELS command with S_ID=hex'00 00 00' and D_ID=hex'FF FF FE'. The command shall be sent using Class 2 service, and support for Class 2 and Class 3 shall be indicated in the service parameters. If the N_Port of the control unit is attached to a Fabric, the response from the Fabric assigns the N_Port of the control unit an N_Port ID (Annex A, "Fabric Address Assignment", describes an address assignment scheme which allows the use of simplified configuration record formats). If an accept response is received, and if support for sequential delivery and Classes 2 and 3 service is indicated, then the control unit shall proceed with further steps in the initialization procedure. If errors prevent the successful receipt of an accept response, or if the service parameters of the response do not indicate support for sequential delivery and Classes 2 and 3 service, further steps in the initialization procedure shall not be performed.

If the N_Port of a control unit receives an FLOGI ELS command before it is able to send an FLOGI ELS command, or if the response to the FLOGI ELS command indicates that the N_Port of a control unit is not attached to a Fabric, then the control unit shall proceed with N_Port login instead of F_Port login.

7.4.2.2 Control Unit N_Port Login

7.4.2.2.1 Control Unit N_Port Login Overview

After the N_Port of the control unit has logged in with the Fabric, or if Fabric login has indicated that the N_Port of the control unit is directly attached to another N_Port, the control unit shall perform N_Port login. When performing N_Port login, Class 2 service shall be used and support for the following features shall be indicated:

a) Classes 2 and 3; and

b) continuously increasing relative offset for solicited data information category.

7.4.2.2.2 Control Unit N_Port Login: Point-to-Point Configurations

If the N_Port of the control unit is directly attached to another N_Port, the N_Port of the control unit shall perform login with the attached N_Port; however, for the case of an emulated control unit, the N_Port of the channel is the same as the N_Port of the control unit, and N_Port login is not repeated on behalf of the N_Port of the emulated control unit. If no errors occur during N_Port login, and if support for Class 2, Class 3, and continuously increasing relative offset for solicited data information category is indicated, then the control unit considers N_Port login successful, and further steps in the initialization process shall be performed. If errors occur, if support for Classes 2 and 3 is not indicated, or if support for continuously increasing relative offset for solicited data information category is not indicated, then control-unit login is regarded as unsuccessful, and further steps in the initialization process shall not be performed.

7.4.2.2.3 Control Unit N_Port Login: Fabric Configurations

If the N_Port of the control unit is attached to a Fabric, then after F_Port login is complete, the N_Port of the control unit shall initiate N_Port login with the Fabric Controller as required in further steps in the initialization process (see 7.4.3 and 7.4.4). When an initialization procedure requires communication with an N_Port with which the control unit is not logged in, the N_Port of the control unit shall initiate N_Port login with that N_Port before performing the procedure. The N_Port of a control unit also initiates N_Port login with a channel when the control unit has received initiative to send a TIN to that channel but the N_Port of the control unit is not logged in with the N_Port of the channel (see 6.4.7). In all other cases, the control unit performs N_Port login only upon receiving a PLOGI ELS request.

If no errors occur during N_Port login, if support for both Class 2 and Class 3 is indicated, and (for the case of login with the N_Port of a channel) if support for continuously increasing relative offset for solicited data information category is indicated, then the control unit shall consider N_Port login successful with respect to that N_Port, and further steps in the initialization process with respect to that N_Port shall be performed; otherwise, the control unit shall consider login with that N_Port unsuccessful, and further steps in the initialization process with respect to that N_Port shall not be performed.

7.4.3 Control Unit Node-Identifier Acquisition

The control unit node-identifier-acquisition procedure provides a means by which the control unit shall acquire the neighboring-node identifier. If the N_Port of a control unit is attached to another N_Port, the N_Port of the control unit shall attempt to acquire the neighboring node identifier by sending an RNID ELS command to the other N_Port as soon as it has completed login with the other N_Port, except in the case of an emulated control unit where node-identifier acquisition is performed by the channel. If the N_Port of the control unit is attached to a Fabric, the N_Port of the control unit shall attempt to acquire the neighboring node identifier by sending an RNID ELS command to the Fabric Controller as soon as it has completed N_Port login with the Fabric Controller, except in the case of an emulated control unit, where node-identifier acquisition is performed by the channel (see 6.3.9 for additional information about the RNID ELS).

When the recipient of the RNID ELS command is able to provide node-identification data of type FC-SB-6, it shall respond with the node-identifier and a flag indicating the validity of the node-identifier. The preferred implementation is for the neighboring node to immediately return a valid node-identifier.

If an LS_RJT is received in response to the RNID ELS, or if errors occur which prevent the control unit from sending the RNID ELS request or receiving a reply containing the requested node-identifier, the control unit shall consider node-identifier acquisition of its neighbor unsuccessful.

The control unit shall continue the initialization process even if the attempt to acquire the node-identifier was unsuccessful. The lack of success in acquiring a valid and current node-identifier shall not prevent the establishment of logical paths or the execution of link-level and device-level functions.

A control unit that checks the node-descriptor flag field shall perform the following actions:

a) When the node-ID-validity code is zero, the control unit shall establish the received 32-byte node descriptor as the node descriptor of its neighbor, and control unit node-identifier acquisition is complete.

b) When the node-ID-validity code is not zero, the control unit shall either:

    1) Check the existing node-descriptor node-ID-validity code if it had previously established the node descriptor of its neighbor, and, if valid and not current, maintain the existing 32-byte node descriptor with an indication that the node-identifier is not current, or maintain an indication that the node descriptor of its neighbor is not valid, or

    2) Not check the existing node-descriptor node-ID-validity code and maintain an indication that the node descriptor of its neighbor is not valid.

When the node-ID-validity code is not zero, the control unit shall retry control unit node-identifier acquisition until a valid node descriptor is acquired. If there is reason to suspect that subsequent retries may not be successful, the node-identification procedure shall be suspended. If the control unit has to provide node descriptors in a function such as link-incident reporting prior to acquisition of a valid and current node descriptor, the control unit shall send the node descriptor of its neighbor with a node-ID-validity code of 1 or 2, as appropriate.

NOTE 21 – When a control unit checks the node-descriptor flag field and recognizes that the node-ID-validity code is not zero, the preferred implementation is option b1 above.

NOTE 22 – When the response to the RNID ELS request contains a value of 2 in the Node-ID-validity field, the control unit should not resend the RNID ELS request so frequently that the frame exchanges seriously degrade the execution of other operations on the link.

NOTE 23 – When a condition occurs that affects the validity of the neighboring-node identifier, the control unit should consider the neighboring-node identifier as not current. This requirement applies for all conditions except for powering off or initial machine loading (IML), in which case, the control unit may discard the neighboring-node identifier.

NOTE 24 – In a preferred implementation involving a Fabric, the Fabric attempts to acquire the node-identifiers of each N_Port which has acquired a node-identifier from the Fabric.

7.4.4 Control Unit State-Change Registration

The control unit state-change-registration procedure enables the N_Port of a channel (in a point-to-point configuration) or a Fabric Controller (in a Fabric configuration) to send registered state change notifications to the control unit (see 6.3.6).

If the N_Port of a control unit is attached to another N_Port, the N_Port of the control unit shall register for state-change notification by sending the SCR ELS command to the other N_Port except in the case of an emulated control unit where state-change registration is performed by the channel, or in the case in which the control unit is already registered to receive state-change notifications. If the N_Port of the control unit is attached to a Fabric, the N_Port of the control unit shall register for state-change notifications by sending an SCR ELS command to the Fabric Controller, except in the case of an emulated control unit, where state change registration is done by the channel, or in the case in which the control unit is already registered to receive state-change notifications. For additional information on the SCR ELS command, see 6.3.7. After the SCR ELS command is accepted, RSCN ELS commands are sent to the N_Port of the control unit when other N_Ports have potentially changed their states. When an accept response to the SCR ELS command is received, the control unit shall consider state-change registration successful, and all IUs containing link-level information which are allowed for a control unit may be sent. If errors occur which prevent the sending the SCR ELS command or receiving an accept response, the control unit shall consider state-change registration unsuccessful and shall not proceed with further steps in the initialization process.

7.4.5 Process login

A control unit that supports FC-SB-6 process login shall perform FC-SB-6 process login when a PRLI ELS request with an FC-SB-6 type code is received from a channel. The control unit responds to the PRLI request with an LS_ACC response with FC-SB-6 process login operating parameters (see 6.3.14), including whether it supports transport-mode operations. When any of the operating parameters provided by the control unit during process login change, the control unit shall send a PRLO to perform explicit process logout (see 6.3.15).

7.4.6 Control Unit Logical-Path Establishment

The control unit shall attempt to establish a logical path when a link-control IU containing a request for the establishment of a logical path between a specified channel image and control-unit image is received. The control-unit-logical-path-establishment procedure consists of either accepting or refusing the requested logical path. If the control unit is able to accept the specified logical path, it shall respond with a link-control IU containing an indication that the request is accepted and the logical path has been established; otherwise, it shall indicate that the logical path is not accepted. When the logical path is established, the control unit shall then allow device-level communication on that logical path between the channel image and control-unit image (see 6.4.2).

The control unit shall consider the logical path to be established upon sending the logical-path-established response without errors. When the control unit considers the logical path to be established, the control unit shall consider the initialization process to be complete for the combination of that channel image and that control-unit image.

A logical path shall no longer be usable for the exchange of IUs containing device-level information when either end of the path no longer considers the logical path to be established. A control unit shall consider a logical path to be no longer established when any of the following events occurs:

a) A remove-logical-path request is received, and an error-free response is returned indicating logical path removed (see 6.4.3);

b) The control unit detects an error for which the recovery action includes removing the logical path (see 11.2); or

c) The control unit is powered off.

When a logical path is not established, the control unit shall not perform device-level functions and protocols with respect to the logical path.

NOTE 25 – A logical path may be lost and reestablished without the recognition of an error condition. When a logical path is not established, the equivalent of a system reset occurs with respect to that logical path. This reset without the recognition of an error condition is acceptable only because there is an explicit requirement for all FC-SB-6-I/O-interface devices to provide reset-event notification (see 9.6). If a logical path is removed, either the program is aware of the reset or the device reports that a reset has occurred. The report occurs only if the logical path is reestablished and an I/O operation is initiated to the device on the logical path.

8 FC-SB-6 Information Units

8.1 FC-SB-6 Information Unit Overview

This standard is based upon the FC-4 Information Unit construct described in FC-FS-4.

Information associated with the execution of an I/O operation and the operation of a device is transferred between the channel and control unit as Information Units. FC-SB-6 Information Units (IUs) contain FC-SB-6 device-level commands, status, data, data descriptor or control information, or FC-SB-6 link-control information. All FC-SB-6 IUs are sent as FC-4 device-data frames (FC-FS-4 routing control bits set to ‘0000’b).

This standard makes use of seven of the FC-FS-4 information categories: the unsolicited-command IU, the command-status IU, the solicited-data IU, the unsolicited-data IU, the solicited-control IU, the unsolicited-control IU, and the data-descriptor IU.

8.2 Rules for Sending FC-SB-6 IUs

8.2.1 Overview of Rules for Sending FC-SB-6 Information Units

This subclause defines the information category a channel or control unit uses when it sends an FC-SB-6 IU, and the placement of the IU within an exchange. All of the allowed information categories and placements within an exchange of FC-SB-6 IUs are summarized in table 8 and table 9. When a particular IU is sent, however, its information category and its placement within an exchange may depend on the conditions under which the IU is sent. For those cases in which such a dependency exists, rules define the conditions under which an information category is used and the conditions under which the IU starts or ends an exchange. Rules for IUs which contain device-level functions are given following the tables. Rules for IUs which contain link-level information are given in 6.4.

Table 8 – Characteristics of IUs Sent by a Channel

FC-SB-6 IU Type	F,M, or L	FC-FS-4 Information Category	Sequence Initiative
Command[1]	F/M	Unsolicited command	H
Command-Data[1]	F/M	Unsolicited command	H
Transport Command [2]	F	Unsolicited command	T/H[3]
Data[1]	M	Solicited data	H
Transport Data[2]	M	Solicited data	T
Control[1]	F	Unsolicited control	H
Control[1]	F /M/ L/FxL	Solicited control	H
Transport Confirm[2]	L	Solicited control	H
Link-Control (request)	F	Unsolicited control	H
Link-Control (response)	FxL/L	Solicited control	H

Key

F:	IU opens an exchange but does not close it
M:	IU is neither first nor last IU of an exchange
L:	IU closes a pre-existing exchange
FxL:	IU opens and closes an exchange
H:	IU Sequence initiative held or last sequence
T:	IU Sequence initiative transferred

Notes

[1] - Sent only in command mode.

[2] - Sent only in transport mode.

[3]- Sequence initiative is held on write if first-transfer-ready disabled; otherwise initiative is transferred.

.

Table 9 – Characteristics of IUs Sent by a Control Unit

IU Type	F,M, or L	FC-FS-4 Information Category	Sequence Initiative
Data[1]	M	Solicited data	H
Transport Data[2]	M	Solicited data	H
Transfer Ready[2]	M	Data descriptor	T
Control[1]	F/M/L/FxL	Solicited control	H
Status[1]	F/M/L/FxL	Solicited data	H
Status[1]	F	Unsolicited data	H
Transport Response[2]	M/L	Command status	T
Link-Control (request)[1]	F	Unsolicited control	H
Link-Control (response)[1]	FxL/L	Solicited control	H

Key

F: IU opens an exchange but does not close it
M: IU is neither first nor last IU of an exchange
L: IU closes a pre-existing exchange
FxL: IU opens and closes an exchange
H: IU Sequence initiative held or last sequence
T IU Sequence initiative transferred or last sequence

Notes

[1] - Sent only in command mode.
[2] - Sent only in transport mode.

8.2.2 Rules for Device-level Functions in Command Mode

a) A channel shall send an IU containing a command or command-data DIB in an unsolicited command IU as the first or middle sequence of an exchange, depending on whether there is an exchange already open for that device. The exchange shall be left open.

b) A status DIB shall be sent as an initiation IU unless sent for a specific device when an inbound exchange already exists for that logical path and device.

c) Status shall be sent as an information category solicited data if sent on an existing exchange or if sent as an initiation IU in direct response to an initiation IU from the channel. The data IU containing the status DIB shall close the inbound exchange unless one of the following conditions apply:

   1) All of the conditions for chaining are satisfied and device-end status is included in the status DIB;

2) The status is sent in response to a control IU;

3) The status is sent as a result of receiving an early end indication; or

4) Supplemental status is included in the status DIB.

Status sent in response to a selective-reset IU requesting retry or unit check with the CI bit set to one shall be regarded as sent in response to the selective-reset IU (a control IU), and the inbound exchange shall be left open.

d) Status shall be sent as an information category unsolicited data if it is sent as an initiation IU and is not a direct response to an initiation IU from the channel. The inbound exchange shall be left open.

e) A control unit shall not send a status DIB for any specific device, on the same logical path, on more than one exchange at the same time.

f) If a channel sends an initiation IU containing a device-level control function or a command to a device on a logical path and simultaneously a control unit sends an initiation IU as information category unsolicited data containing status for the same device on the same logical path, then the control unit shall send a response to the initiation IU using the existing inbound exchange with an IU which contains the same token (see 8.5.5). as the initiation IU from the channel. The channel shall discard the initiation IU from the control unit and shall wait for another IU on the inbound exchange from the same device which contains a response to its initiation IU.

8.2.3 Rules for Device-level Functions in Transport Mode

a) A channel shall send a transport-command IU as an unsolicited command IU as the first sequence of a transport exchange. The exchange shall be left open and sequence initiative for the transport exchange shall be transferred to the recipient unless the operation is a write operation and first-transfer ready is disabled is in effect (see 9.3.2.2).

b) For a write operation when first-transfer ready disabled is in effect, the channel may send a write transport-data IU immediately following the transport-command IU. The transport-data IU shall not contain an amount of data that exceeds the amount of data allowed for a first write-transport-data IU when first-transfer ready disabled is in effect (see 9.3.2.2). The exchange shall be left open and sequence initiative for the transport exchange shall be transferred to the recipient.

c) Status shall be sent in a transport-response IU as information category command status in response to a transport-command IU from the channel. The transport-response IU containing the status shall close the transport exchange unless confirm completion (see 8.17) is required for the transport-response IU. When confirm completion is requested, the exchange shall be left open and sequence initiative for the transport exchange shall be transferred to the channel.

d) When a transport exchange is open for a specific device and logical path, the channel may send an unsolicited transport-command IU as the first sequence of another transport exchange to the same device and logical path if the new transport-command IU specifies the interrogate operation (see 9.3.3.2).

e) If a channel sends a transport-command IU to a device on a logical path and simultaneously a control unit sends an initiation IU as information category unsolicited data containing status on another exchange for the same device on the same logical path, then the following actions shall be performed in the order listed by the channel and control unit:

1) The channel shall respond to the initiation IU from the control unit with a stack status control IU on a new outbound exchange;

2) If the transport-command IU is for a write operation and first-transfer-ready is disabled, the channel shall send a transport-data IU to the control unit;

3) At this point, sequence initiative for the transport exchange is held by control unit and the channel waits for an IU from the CU sent in response to the command IU; and

4) The control unit shall respond to the stack status control IU with a DACK control IU on the existing inbound exchange and then shall either perform the transport operation specified by the transport-command IU from the channel or presents the stacked status with busy status in the transport-response IU on the transport exchange.

8.3 FC-SB-6 IU Structures

FC-SB-6 IUs used to perform operations in command mode use a uniform IU structure as shown in figure 16 and described in the following sections. The format of FC-SB-6 IUs used to perform I/O operations in transport mode are shown in figure 17 and described in the following sections.

Command mode FC-SB-6 IUs all contain an 8-byte FC-SB-6 header followed by an 8-byte IU header. Immediately following the IU header is a field referred to as the device-information block (DIB). A DIB shall contain a minimum of 16 bytes consisting of a 12 byte DIB header followed by four bytes of longitudinal-redundancy check (LRC - see 8.6.3). For certain DIB types, a variable length DIB data field may immediately follow the LRC. The maximum length of the DIB data field shall be 8160 bytes. Six different DIB types are defined; they are: the data DIB, command DIB, status DIB, control DIB, command-data DIB, and link control DIB. The type and structure of the DIB is determined by bits in the information unit identifier (IUI) field of the IU header (see 8.5.2).

There are five types of FC-SB-6 IUs defined for transport-mode operations: transport-command, transport-response, transport-data, transfer-ready and transport-confirm IUs. FC-SB-6 transport-command and transport-response IUs contain the 8-byte FC-SB-6 header; the transport-data IU, the transfer-ready IU and the transport-confirm IU do not contain an FC-SB-6 header and rely on the fully qualified exchange identifier (FQXID) associated with the transport exchange to identify the logical path and device. The FQXID is provided in the FC-FS-4 header and is composed of the source port identifier, the destination port identifier, the OX_ID field value, and the RX_ID field value (see FC-FS-4).

FC-SB-6 general IU payload structure for command-mode IUs

FC-SB-6 Header	IU Header	DIB Header	LRC	DIB Data
(8)	(8)	(12)	(4)	(0-8160)

FC-SB-6 unsolicited command IU with command DIB

FC-SB-6 Header	IU Header	Command Header	LRC
(8)	(8)	(12)	(4)

FC-SB-6 unsolicited command IU with command-data DIB

FC-SB-6 Header	IU Header	Command Header	LRC	Data	Pad	CRC
(8)	(8)	(12)	(4)	(0-8160)	(0-3)	(0/4)

FC-SB-6 solicited data IU with data DIB

FC-SB-6 Header	IU Header	Data Header	LRC	Data	Pad	CRC
(8)	(8)	(12)	(4)	(0-8160)	(0-3)	(0/4)

FC-SB-6 solicited/unsolicited data IU with status DIB

FC-SB-6 Header	IU Header	Status Header	LRC	Supplemental Status	Pad	CRC
(8)	(8)	(12)	(4)	(0-32)	(0-3)	(0/4)

FC-SB-6 solicited/unsolicited control IU with control DIB

FC-SB-6 Header	IU Header	Control Header	LRC	Control Payload	Pad	CRC
(8)	(8)	(12)	(4)	(0-256)	(0-3)	(0/4)

FC-SB-6 solicited/unsolicited control IU with link control DIB

FC-SB-6 Header	IU Header	Link Header	LRC	Link Payload	Pad	CRC
(8)	(8)	(12)	(4)	(0-8156)	(0-3)	(0/4)

Legend: see following page.

Figure 16– IU Payload Structures for Command-Mode IUs

FC-SB-6 transport-command IU

FC-SB-6 Header	Transport Cmd Header (TCH)	Transport Cmd Area Header (TCAH)	Transport Command Area (TCA)	LRC	DL
(8)	(4)	(16)	(8-240)	(4)	(4)

FC-SB-6 transport-data IU (read transfer)

Data	Final Pad	Final CRC
	(0-3)	(0/4)

total length = 0 - (4 gigabytes minus 4[1])

FC-SB-6 transport-data IU (write transfer)

CRC or Extended CRC Offset Block[2]	COB or eCOB Pad	COB or eCOB CRC	TCA Extension[3]	TCA X Pad	TCA X CRC	Data	Intermediate PAD[4]	Intermediate CRC[4]	Data	Final Pad	Final CRC
(0-N)	(0/4)	(0/4)	(0-N)	(0/4)	(0/4)		(0-3)	(0/4)		(0-3)	(0/4)

total length = 0 - (4 gigabytes minus 4[1])

FC-SB-6 transport-response IU

FC-SB-6 Header	Status	LRC	Extended Status	LRC
(8)	(20)	(4)	(0/(32-64)	(0/4)

FC-SB-6 transfer-ready IU

Relative Offset	Maximum Burst Length	Reserved
(4)	(4)	(4)

FC-SB-6 transport-confirm IU - no payload

null

(0)

Legend:
(x-y) - number of bytes in field may range from x bytes to y bytes.
(x/y) - number of bytes in field is either x bytes or y bytes.

Notes:
[1] See 9.3.2 for restrictions on the data byte count for read and write transfers.
[2] See 8.13.5.2.1.2 for a description of when the CRC Offset Block (COB) or extended CRC Offset Block (eCOB) fields are provided.
[3] See 8.13.5.2.1.3 for a description of when the TCA Extension (TCAX) fields are provided.
[4] Multiple intermediate pad and CRC words may be provided in write transport-data IUs during an operation that performs write data transfer. For an operation that performs read data transfer, a single CRC word is provided for read data transferred in all read transport-data IUs.

Figure 17 – IU Payload Structures for Transport-Mode IUs

8.4 FC-SB-6 Header

8.4.1 FC-SB-6 Header Overview

An FC-SB-6 header shall be provided in all command-mode FC-SB-6 IUs and in the transport-command and transport-response IUs. An FC-SB-6 header is not provided in the transport-data, transfer-ready and transport-confirm IUs. Although the transport-response IU contains an FC-SB-6 header, the contents of the header are not used to identify the operation. In transport mode, the FQXID (see 4.7.4) in the transport-command IU that initiates a transport exchange is used to identify all other IUs for the exchange.

8.4.2 FC-SB-6 Header Format

The FC-SB-6 header provides FC-4 addressing information to identify the logical path and the device for the exchange. figure 18 shows the FC-SB-6 header format.

0	Reserved/x'FE'	CH Image ID	Reserved	CU Image ID
1	Device Address		Reserved/Model Dependent	
	0	8	16	24

Figure 18 – FC-SB-6 Header

Bits 16-23 of word 0 shall be reserved. Bits 0-7 of word 0 and bits 16-31 of word 1 shall be reserved in all command-mode IUs. In transport-command and transport-response IUs, bits 0-7 of word 0 shall be set to hex'FE'. In transport-command IUs, bits 16-31 of word 1 shall be set to a model dependent value; in transport-response IUs, bits 16-31 of word 1 shall be set to bits 16-31 of word 1 of the command IU for which the response IU is being sent.

8.4.3 Channel Image ID

Bits 8-15 of word 0 contain the 8-bit ID of the channel image. For all IUs except a control IU containing a link-control DIB, the channel image ID specified shall correspond to a logical path previously initialized by the establish-logical-path procedure; otherwise a logical-path-not-established error shall be recognized. For a link-control DIB, the channel-image ID shall be meaningful for the following link-control functions: establish-logical path (ELP), logical-path established (LPE), remove-logical path (RLP), logical-path removed (LPR), test-initialization (TIN), test-initialization result (TIR), link-level reject (LRJ), and link-level busy (LBY). For the LACK link-control function the channel-image ID shall be set to zero by the sender and ignored by the recipient.

8.4.4 Control-Unit Image ID

Bits 24-31 of word 0 contain the 8-bit ID of the control-unit image. For all IUs except a control IU containing a link-control DIB, the control-unit image ID specified shall correspond to a logical path previously initialized by the establish-logical-path procedure; otherwise a logical-path-not-established error shall be recognized. For a link-control DIB, the control-unit image ID shall be meaningful only for the following link-control functions: establish-logical path (ELP), logical-path established (LPE), remove-logical path (RLP), logical-path removed (LPR), test initialization (TIN), test initialization result (TIR), link-level reject (LRJ), and link-level busy (LBY). For the LACK link-control function the control-unit image ID shall be set to zero by the sender and ignored by the recipient.

8.4.5 Device Address

When the AS bit in the IU header of a command-mode IU is set to one, and for all transport-command and transport-response IUs, bits 0-15 of word 1 shall contain the address of the device for the exchange. The device address specified shall correspond to a device configured to the control-unit image specified by the control-unit image ID; otherwise, an address exception condition shall be recognized by the control unit.

Bits 0-7 of the device address shall be set to zeros by the source of an IU. In command mode, if bits 0-7 of the device address are not set to zeros when the AS bit is set to one, a device-level protocol error shall be recognized. When the AS bit is set to zero, the 16-bit device-address field shall be set to zero by the sender of the IU and is ignored by the recipient of the IU. In transport mode, if bits 0-7 of the device address are not set to zeros, a device-level protocol error shall be recognized.

Only one device address shall be used during a single connection. Once a device address has been identified for a connection, the use of a different device address during this same connection shall result in a device-level protocol error being recognized.

If a control unit receives either a command-mode IU with the AS bit set to one or a transport-command IU with a device address of a not-ready device, it shall either perform the specified function and provide the appropriate response, if any, or, when the specified function requires a ready device, generate unit-check status; the sense data associated with the unit check shall indicate intervention required. If a control unit receives an IU with the AS bit set to one or a transport command or status IU with and the device address of an uninstalled device, it shall generate either an address-exception condition or, optionally, unit-check status (see 8.11.2.10).

The manner in which device addresses are assigned is model dependent.

8.5 IU Header

8.5.1 IU Header Format

The information unit (IU) header is provided in all command-mode IUs and immediately follows the FC-SB-6 header and precedes the DIB header. The IU header, shown in figure 19, provides FC-SB-6 control flags and necessary information in order to associate an IU to a specific CCW.

<table>
<tr><td>0</td><td>IUI</td><td>DH Flags</td><td colspan="2">CCW Number</td></tr>
<tr><td>1</td><td>Reserved</td><td colspan="3">Token</td></tr>
<tr><td></td><td>0</td><td></td><td>16</td><td>31</td></tr>
</table>

Figure 19 – IU Header

The IU header consists of five fields, the information-unit identifier (IUI) field, device header (DH) flags field, CCW number field, a reserved field, and the token field. Byte 0 of word 1 shall be reserved and set to zero by the sender and ignored by the recipient.

8.5.2 Information-Unit Identifier

8.5.2.1 Information-Unit Identifier Format

Byte 0 of word 0 of the IU header is the information-unit identifier (IUI). The IU type and the functions that affect field formats and interpretation are identified by bits in the IUI, which have the format shown in figure 20.

0	0	0	AS	ES	T3	T2	T1
0	1	2	3	4	5	6	7

Figure 20 – IU Identifier

Bits 0-2 of the IUI shall be reserved and are set to zeros by the sender for all IU types and ignored by the recipient.

8.5.2.2 Address Specific (AS)

The AS bit, bit 3 of the IUI, when set to one, shall indicate that the IU is associated with the specific device identified by the device-address field of the FC-SB-6 header. When the AS bit is set to zero, the IU shall not be associated with a specific device, and the device address shall not be used.

For a data IU containing a data DIB or command IU, the AS bit shall be set to one; otherwise, a device-level protocol error shall be detected.

For a data IU containing a status DIB, the AS bit may be set to either one or zero, depending on whether the status is associated with the device or the control unit (see table 20).

For a control IU, the AS bit may be either one or zero, depending on the particular control function (see 8.11, table 20, and 6.4).

8.5.2.3 Supplemental Status (ES)

The ES bit, bit 4 of the IUI, may be set to either one or zero. The meaning of the bit when set to one depends on the IU type.

For a data IU containing a data DIB or a command IU the ES bit shall be set to zero by the sender and ignored by the recipient.

For a data IU containing a status DIB, the ES bit may be set to either one or zero. If the ES bit is set to one, supplemental status, sometimes referred to as extended status, shall be present in the supplemental status field. If the ES bit is set to zero, the length of the supplemental status field shall be zero. The ES bit may be set to one only when the status DIB contains unit check status.

For a control IU specifying the status-accepted function, the ES bit may be set to either one or zero. If the ES bit is set to one, the channel shall be indicating that it has accepted the supplemental status that was presented along with the status byte. If the ES bit is set to zero in a status-accepted IU that is sent in response to a status DIB with the ES bit set to a one, it shall indicate that the status has been accepted, but the supplemental status has not been accepted (see 9.2.3.2). The ES bit shall be set to one in a status-accepted IU only in direct response to status with the ES bit set to one. For all other control functions, the ES bit shall be set to zero by the sender and ignored by the recipient.

8.5.2.4 DIB Type (T3,T2,T1)

Bits 5-7 of the IUI define the type of DIB contained in the IU. The DIB type also determines the format of the DIB. Table 10 summarizes the setting of the T3-T1 bits.

Table 10 – DIB-Type Settings

T3	T2	T1	DIB Type	DIB Format
0	0	0	Data	Data
0	0	1	Command	Command Header
0	1	0	Status	Flags, Status, Count, Supplemental Status
0	1	1	Control	Control Function and Parameters
1	0	0	Command-data	Command Header and Data
1	0	1	Link Control	Link-control Functions
1	1	0	Reserved	
1	1	1	Reserved	

A data DIB shall be sent only in a data IU and shall be received by either the channel or control unit.

A command DIB shall be sent only in a command IU and shall only be received by a control unit. If a channel receives a command DIB, a device-level protocol error shall be detected.

A command-data DIB shall be sent only in a command IU and shall only be received by a control unit If a channel receives a command-data DIB, a device-level protocol error shall be detected.

A status DIB shall be sent only in a data IU and shall only be received by the channel. If a control unit receives a status DIB, a device-level protocol error shall be detected.

A control DIB shall be sent only in a control IU and shall be received by the channel or control unit.

A link-control DIB shall be sent only in a control IU and shall be received by the channel or control unit.

8.5.3 Device-Header Flags

8.5.3.1 Format of Device-Header Flags

Byte 1 of word 0 of the IU header is the device-header flags (DHF) field. The device-header flag bits are used individually or collectively to invoke certain FC-SB-6 protocols to control the execution of an I/O operation.

The format of the device-header flags field and the description of each of its flag bits follow. Bits 1,2,6, and 7 of the device-header flag field shall be reserved and are set to zeros by the sender and ignored by the recipient. The format of the device-header flags field is shown in figure 21.

E	0	0	CH	EE	CNP	0	0
0	1	2	3	4	5	6	7

Figure 21 – Device-Header Flags

8.5.3.2 End (E)

The E bit, bit 0, when set to one for either a command-data DIB or data DIB, shall be used to indicate that the data sent exactly satisfies the CCW count and that the CRC field is present in the DIB. Also, the E bit, when set to one for a status DIB, shall be used to indicate that the quantity of data sent by the channel with an early end indication exactly satisfied the record length (see 8.5.3.4). The CCW number and token in the IU header, collectively and in a model dependent manner, identify the CCW to which the E bit applies.

For a data DIB, the E bit may be set to either one or zero. If the E bit is set to one, the CRC field shall be present in the data DIB and the total quantity of data sent for the CCW shall exactly satisfy the CCW count. If the E bit is set to zero, and the data sent for the CCW does not exactly satisfy the CCW count, the presence of the CRC field shall be determined by the EE bit.

For a command-data DIB, the E bit may be set to either one or zero. If the E bit is set to one, the CRC field shall be present in the command-data DIB and the data in the IU exactly shall satisfy the CCW count. If the E-bit is set to zero, and the data in the command-data DIB does not exactly satisfy the CCW count, the presence of the CRC field shall be determined by the EE bit.

If the DIB type being sent is a command DIB for a write operation and the CCW count is zero, the E bit shall be set to one. If the DIB type being sent is a command DIB for a read operation, the E bit shall be set to zero.

For a status DIB, the E bit may be set to one or zero. The E bit shall be set to one for a status DIB to indicate that for a write operation with an early end indication, the data received from the channel exactly satisfied the record length; otherwise, the E bit shall be set to zero.

For a control DIB, the E bit shall be set to zero by the sender and ignored by the recipient (see 8.11 and table 20).

For a link-control DIB, the E-bit may be set to one or zero. The E bit shall be set to one when a link payload including the CRC field is present.

The presence of the CRC field does not mean that a valid CRC is provided. The CNP bit shall be checked to determine if a valid CRC is provided in the CRC field.

8.5.3.3 Chaining (CH)

The CH bit, bit 3, when set to one, is used by the channel to signal its intention to chain or to confirm that chaining is continuing.

For a data IU, the CH bit shall be set to zero by the sender and is ignored by the recipient.

For a command IU, the CH bit may be set to either one or zero. If the CH bit is set to one, the command IU shall be a command update as a result of either command chaining or data chaining and the DIB shall contain information from the current CCW. Whether the command update is for data chaining or command chaining shall be indicated by the DU bit in the command flag field. If the CH bit is set to zero, the command IU shall not be a command update. The CH bit shall be set to zero in a command IU initiating the channel program; otherwise, a device-level protocol error shall be detected. The CH bit shall be set to one in all remaining command IUs received during the execution of the channel program; otherwise, a device-level protocol error shall be detected.

For a control IU from the channel, the CH bit may be set to one only when the status-accepted function is indicated either during command chaining and the status does not include device end or when the status represents a retry request for an I/O operation and the status does not include device end. The CH bit shall be set to one by the channel when chaining is still indicated for the I/O operation and set to zero when chaining is not or no longer indicated for the I/O operation or when a retry request is not accepted. For all control IUs specifying other than the status accepted function, the CH bit shall be set to zero by the sender and ignored by the recipient.

8.5.3.4 Early End (EE)

The EE bit, bit 4, when set to one for either the control-end function in a control DIB, a data DIB, or a command-data DIB shall be used to indicate that the quantity of data sent for the CCW indicated by the CCW number is less than the CCW count specified for that CCW and for the data DIB and command-data DIB, the CRC field is present (see 8.11.2.3). If the EE bit is set to zero, an early end condition is not being indicated and the presence of the CRC field shall be determined by the E bit.

When, for a write operation, the channel is able to send some, but not all, of the data specified for a CCW, the EE bit shall be set to one for either the command-data DIB or data DIB containing the CRC field. The CRC field may be present either in the same DIB containing data or in a data DIB containing no data immediately following the IU containing the last data bytes. If, for a CCW with a nonzero CCW count, the channel is unable to send any data, the EE bit shall be set to one for a command-data DIB which contains the CRC field. The CRC value sent shall be the initialized value of the CRC generator. See 8.6.5 for additional information.

When during data chaining the channel is unable to obtain the next CCW and perform the command update, the EE bit shall be set to one in a control-end IU in order to indicate that an early end condition was detected and the channel is unable to send or receive any more data for the current command.

When, for a read operation, the control unit is able to send some, but not all, of the data specified for a CCW, the EE bit shall be set to one in the data DIB containing the CRC field. The CRC field may be present either in the same DIB containing data or in a data DIB containing no data immediately following the IU containing the last data bytes.

Table 11 summarizes the use of the EE bit for CCWs with a nonzero CCW count:

Table 11 – EE-bit Table

Conditions	Sent by	Number of Bytes Sent	DIB Type	EE	CRC Field
All write CCWs. Some data sent	CH	< CCW Count	Command-data or data	1	Yes
All write CCWs No data sent	CH	0 Bytes	Command-data	1	Yes
Data chain CCW - Read or write No command update sent	CH	0 Bytes	Control	1	No
All read CCWs Some data sent	CU	< CCW Count	Data	1	Yes

When an early end indication is received by a control unit, status shall be returned at the completion of the operation for which the early end indication was sent.

When the Early End condition is recognized, the control unit shall disregard detection of an incorrect-length condition, if any, and shall depend on the channel to make the decision to continue chaining or not (see 8.7.2.3).

For a status DIB in a data IU and control IU specifying a control function other than the control-end function, the EE bit shall be set to zero by the sender and ignored by the recipient.

The EE bit and the E bit shall never be both set to one in the same IU and if both are found set to one in an IU for which both are meaningful, a device-level protocol error shall be recognized.

8.5.3.5 CRC Not Provided (CNP)

The CNP bit, bit 5, when set to one for a data, command-data, status, control, or link-control DIB shall indicate that the value contained in the CRC field is not valid, and that CRC checking for the data associated with the CRC field should not be performed. The CNP bit shall only have meaning when the CRC field is present (see 8.6.5). When the CNP bit is set to zero, the CRC field shall contain a valid value, and if the receiver provides CRC checking, and the receiver's calculated value does not equal the received value, an FC-SB-6 CRC error shall be detected.

8.5.4 CCW Number

The CCW number, bytes 2 and 3 of word 0, of the IU header contains a model-dependent 16-bit binary value that is assigned by the channel to the CCW associated with the IU being sent.

A CCW number for a command IU initiating an I/O operation shall not be reused in a subsequent command IU until the command which initiated the I/O operation is no longer retryable (see 9.5.2). A CCW number for a command IU sent during data chaining and following a command IU initiating an I/O operation shall not be reused in a subsequent command IU until the channel has recognized that the CCW corresponding to this CCW number has been executed. This occurs when a data or control IU has been received indicating that execution of the channel program has progressed to a CCW contained in a subsequently transferred command IU. See 9.2.2.4 for additional information on data chaining.

The CCW number uniquely identifies a CCW specifying a retryable command or a CCW not specifying a command sent during data chaining which the channel has not yet recognized as executed.

The value hex’0000’ is not a valid CCW number and shall not be used by the channel or control unit to identify a CCW. This value shall be used in those IUs that do not specify a specific CCW number. All values other than hex’0000’ shall be valid for assignment by the channel to CCWs executed. Whenever a value other than hex’0000’ is present in the CCW number field, a valid token shall also be present in the token field in word 1 (see 8.5.5). If the channel receives an IU with the value hex’0000’ in the CCW number field and that IU is required to have a valid CCW number, a device-level protocol error shall be recognized.

A control unit shall not check the CCW number.

Command IUs and data IUs containing a data DIB shall require a valid CCW number. The channel shall include the CCW number from the command IU in all data IUs sent to satisfy the CCW count specified by the command IU, and the control unit shall include this CCW number in all data IUs sent to satisfy the CCW count specified by the command IU.

Some IUs do not pertain to a CCW, and therefore, do not require a valid CCW number. For example, an unsolicited data IU containing a status DIB for asynchronous status or a link control request IU.

When a disconnection occurs as a result of the acceptance of status by the channel, all CCW numbers assigned to CCWs associated with the connection being removed shall become void, and as such shall be available for re-use by the channel. If the control unit subsequently reconnects to present status, a CCW number of hex’0000’ shall be used. If a disconnection occurs as a result of an abnormal condition and a system reset or selective reset is not performed, the CCW numbers being maintained at the time of disconnection shall be preserved. The CCW numbers shall be preserved at least until the device-level recovery performed after the disconnection has completed and may remain in effect afterward, depending on the result of the recovery action.

The channel may indicate in a control IU the CCW number for which the control function is to be performed. If the channel sends a selective reset control IU with the RO bit, RU bit, or RC bit set to one, then a valid CCW number shall be required if the request is for a specific command; otherwise, a CCW number of hex’0000’ shall be used.

When the control function specified does not require a valid CCW number to be specified, the value hex’0000’ shall be used (see the individual control function descriptions for CCW number requirements).

For a status DIB, a valid CCW number may or may not be required depending on the status being presented. If the status DIB is for the presentation of channel-end status during an existing connection, status in response to a command received in a command IU, or stacked or pending status with the busy indication in response to a command IU, the CCW number used shall be the CCW number from the command IU. If the status DIB is for the presentation of device-end status after a disconnection, the CCW number used shall be hex’0000’. If the status DIB is sent in direct response to a request-status IU that is used to retrieve stacked status, the CCW number from the request-status IU shall be used. If the status DIB is in response to a channel initiated recovery procedure for a specific CCW number, then the CCW number from the recovery request from the channel shall be used. If a control unit sends retry status in an unsolicited data IU, the CCW number used shall be the CCW number from the command IU that is to be retried. For all other cases where a status DIB is sent, a valid CCW number shall not be required and the value hex’0000’ shall be used.

8.5.5 Token

The token field, bytes 1-3 of word 1, of the IU header shall contain a 24-bit binary value that is provided by the channel. The value hex’000000’ is not a valid token and shall be used only in those IUs that do not require a valid token. All other values of the token are valid and may be used by the channel. The method of determining the token value to be sent in an IU and its use by the channel is model dependent.

The token field shall contain a valid token, whenever the IU contains a valid CCW number. If the token field contains a value of all zeros, the CCW number is meaningless, and shall be set to all zeros by the sender and ignored by the receiver. When the token field contains a valid token, the CCW number field may or may not contain a valid CCW number, depending on other information associated with the IU.

A control unit shall not check the token.

A valid token shall always be sent in a command IU and data IU containing a data DIB. The control unit shall include the token from the command IU in all data IUs sent to satisfy the CCW count specified by the command IU. If the channel detects that a valid token is not present when it receives a data DIB, a device-level protocol error shall be recognized.

A valid token shall be sent by the channel in an unsolicited control IU. Examples are: system reset, cancel, and link control requests, ELP or TIN.

A solicited control IU, sent by a control unit shall contain the token from the IU to which it is being sent in response. A solicited control IU, sent by a channel shall contain a valid token.

A valid token shall be sent in a solicited data IU containing a status DIB. If the status DIB is for either the presentation of channel-end status during an existing connection, status in response to a command received in a command IU, or stacked or pending status with the busy indication in response to a command IU, the token used shall be the token from the command IU. Examples are: any status containing channel-end status and control unit busy status. If the channel receives a solicited data IU with a status DIB with a token that is not expected, or a token of all zeros, a device-level protocol error shall be recognized.

The invalid token (hex’000000’) shall be sent in an unsolicited data IU containing a status DIB with asynchronous status and in an unsolicited control IU sent by a control unit. Examples are: asynchronous status presentation and the link-control request, TIN.

8.6 Device Information Block (DIB) Structure

8.6.1 DIB Structure Overview

A DIB is provided in all command-mode IUs and consists of a DIB header, LRC, and for some DIB types a DIB-data field. The DIB type is identified by bits in the IU header. Figure 22 shows the DIB structure common to all DIB types.

DIB Header	LRC	DIB Data
(12)	(4)	0 - 8160

Figure 22 – Basic DIB Structure

8.6.2 DIB Header

8.6.2.1 DIB Header Format

The DIB header is the first 12 bytes of every DIB type and immediately follows the IU header. The contents of the DIB header and the DIB data field are determined by the DIB type. Figure 23 shows the basic DIB header structure common to all DIB types.

0	DIB Type Specific		
1	DIB Type Specific		
2	Reserved	IU Count	DIB Data Byte Count
	0 7	15	31

Figure 23 – DIB Header Structure

The format and contents of the DIB header fields are described in other sections in this clause. The following sections describe those fields in a DIB that are common to more than one DIB type.

8.6.2.2 IU Count

Byte 1 of word 2, of each DIB header type shall contain the IU count. The IU count is an 8-bit binary number that shall be included in every IU sent during an open exchange.

The IU count value shall be initialized to hex’00’ when an exchange is opened and the first IU sent. The value of zero shall be inserted in the IU count field of the DIB header for the sequence sent to open an exchange. After each IU is sent on a given exchange, the IU count shall be advanced by one and the new value included in the IU count field of the next IU sent. When the IU count reaches hex’FF’ and additional sequences are to be sent, the IU count shall be permitted to wrap and the next value used is hex’00’.

When an exchange is closed, there is no further need for an IU count parameter to be maintained by the channel or control unit for that exchange.

If an IU is received with an IU count value that is not one greater than the IU count value received in the previous IU received for a given exchange, or, if the previous IU count was hex’FF’ and the next IU count received is not hex’00’, an IU count error shall be recognized and the IU shall be discarded. If an IU is received that opens an exchange, it shall have an IU count value of zero, otherwise an IU count error shall be recognized and the IU shall be discarded. When an IU is discarded because of an IU count error, the recovery performed depends on the operation or function being performed. In some cases a retry of the operation may be attempted and in other cases the operation or function may be terminated without retry.

8.6.2.3 DIB Data Byte Count

Bytes 2 and 3 of word 2, of the DIB header shall contain the DIB data byte count. The DIB data byte count is a 16 bit binary number that represents the number of bytes of data contained in the DIB-data field. Pad bytes and CRC shall not be included in the DIB data byte count. For a command DIB, the DIB data byte count shall be set to zero by the channel, and ignored by the control unit.

For a status DIB, the DIB data byte count shall be valid only when the ES status flag bit is set to one. When the ES bit is set to one, the DIB data byte count shall represent the number of bytes of supplemental status in the DIB data field. When the ES bit is set to zero, the DIB data byte count shall be ignored.

For a link control DIB, the DIB data byte count shall be valid only when the TIR link-control function is indicated. The DIB data byte count shall represent the number of bytes of link payload in the DIB data field.

For a control DIB, the DIB data byte count field shall be valid only when the Purge-Path-Response function is sent in response to a Purge Path request with a non-zero error code, and error-code transfer is supported.

8.6.3 Longitudinal-Redundancy-Check Field

The longitudinal-redundancy-check (LRC) field shall contain a 32-bit redundancy-check code, immediately following the DIB header in bytes 28 to 31 of the IU (see figure 16). LRC shall be provided only on the bytes contained in the FC-SB-6 header, IU header, and DIB header.

The sender of an IU shall generate LRC and include the value calculated in the LRC field of that IU. The recipient of an IU shall also generate LRC and compare the value calculated to the LRC contained in the IU. If the value of the LRC contained in the IU equals the value calculated by the recipient, a valid LRC shall be recognized; otherwise, an invalid LRC shall be recognized. If a valid LRC is recognized, the contents of the FC-SB-6 header, IU header, and DIB header shall be considered valid; otherwise, the contents of these headers shall be considered invalid and an LRC error shall be recognized.

See Annex C for a description of the procedure and an example of the LRC calculation.

8.6.4 DIB Data Field

8.6.4.1 DIB Data Field Format

The DIB Data field, if present, is a variable length field which contains either data, data and pad bytes, data and a CRC field, data and pad bytes and a CRC field, or just a CRC field. The quantity of data contained in the DIB data field shall be indicated in the DIB data byte count field of the DIB header. A command DIB and a control DIB shall always be sent with a DIB data field of zero bytes. The maximum number of bytes that shall be sent in the DIB data field shall be 8160 bytes. The format of the DIB data field is shown in figure 24.

DIB Data	Pad	CRC

Figure 24 – DIB Data Field

When the data field contains the maximum of 8160 bytes of data, the pad field and CRC field shall both contain 0 bytes. When the CRC field is present in the DIB, the maximum number of bytes in the

DIB data field and pad shall be 8156. All DIB data fields sent for a CCW shall contain 8160 bytes with the possible exception of the last data DIB sent for the CCW. See 8.8 and 8.9 for additional information.

8.6.4.2 Pad

Pad bytes, if present, are contained in the last word of the data field and shall be used to pad the data field to the next word boundary.

For a command-data DIB or a data DIB, pad bytes shall be present only in the last IU containing data that is sent for a CCW and then only when boundary alignment is necessary. For a command IU with a command-data DIB, pad bytes shall be present only if all of the data to be sent for the current CCW is included and word alignment is necessary. For a status DIB with supplemental status, a control DIB with a control payload, and a link-control DIB with a link payload, pad bytes shall be present only if word alignment is necessary.

The value used for a pad byte is model dependent.

Pad bytes, if present, shall be included in the generation of CRC but shall not be included in the byte count for the DIB-data field.

8.6.5 Cyclic-Redundancy-Check Field

When CRC generation is provided by the sender, the cyclic-redundancy-check (CRC) field, when present in an IU, shall contain a word-aligned 32-bit redundancy-check code, and the CNP bit shall be set to a zero. When CRC is not provided, the word-aligned CRC field shall contain a model-dependent value, and the CNP bit shall be set to a one. The requirements for the presence and location of the CRC field shall be the same, independent of whether CRC generation is provided or not.

The CRC field may be present in a command-data DIB, data DIB, status DIB, control DIB, and link control DIB. For a command-data DIB and a data DIB, the CRC field shall be present only when either the E bit or the EE bit in the device-header flags field is set to one. For a status DIB, the CRC field shall be present only when the supplemental status (ES) bit is set to one. For a control DIB, the CRC field shall be present only if a control payload is included. For a link control DIB containing a link payload, the CRC field shall be present and the E bit is set to one.

CRC shall be generated on data, supplemental status, control, and link payloads that are sent in the DIB-data field of an IU. When data is sent, CRC shall be generated on the data associated with a CCW. If more than one CCW is used to transfer the data associated with a command (data chaining), CRC shall be generated and sent with the data for each CCW, that is, CRC generation shall not span the data associated with more than one CCW. If it is possible to send the data associated with a CCW in one IU, then the CRC for that data shall be sent as the last four bytes of the DIB of that IU. If it is not possible to send the data associated with a CCW in one IU, then the CRC for that data shall be sent as the last four bytes of the last data IU required. In some cases it may be necessary to send an IU with a DIB containing only a DIB header and a CRC, as when either the remaining quantity of data for a CCW to be sent exceeds 8156 bytes but is less than or equal to 8160 bytes, or an early end condition occurs and CRC is not included with the last data sent for the CCW.

When CRC generation is provided, the sender shall generate CRC on the data sent and insert the value calculated at the end of the last data DIB. When CRC checking is provided and the CNP bit is set to zero, the recipient shall generate CRC on the data received and compare the value calculated to the CRC contained in the IU. If the value of the CRC contained in the IU equals the value calculated by the recipient, a valid CRC check shall be recognized; otherwise, an FC-SB-6 CRC error shall be recognized. If the receiver does not provide CRC checking or the CNP bit is set to one, the contents of the CRC field shall be ignored.

Whenever some, but not all, of the data from a CCW is used (such as when a cancel is received during an I/O operation), and CRC checking is provided, and the CNP bit is set to zero, that data shall be verified by checking the CRC for the entire CCW.

The processing of CRC, in terms of generation and checking, shall follow the equations in Annex A of FC-FS-4; however, the coefficients of the polynomial representing the DIB data field are not chosen according to bit transmission order specified in Annex A. The coefficient of the highest order term of F(x), which is the polynomial representing the DIB data field, shall represent the most-significant bit of the DIB data field; lower-order coefficients of the polynomial F(x) correspond to less-significant bits of the DIB data field in sequential order. The 32-bit CRC shall be the 32 coefficients of the frame check sequence (FCS) polynomial in Annex A, equation (1). The most significant bit of the CRC shall be the coefficient of the highest order term of the FCS polynomial; sequentially less-significant bits shall be the coefficients of sequentially lower order terms of the FCS polynomial. The CRC is transmitted on the link in the same bit-transmission order as all other words of the DIB data field. Additional information may be found in Appendix B of *Fiber Distributed Data Interface - Media Access Control (FDDI-MAC).*

When optional CRC generation initialization is not being used, the initialized value of the CRC generator shall be hex’FFFFFFFF’.

When optional CRC generation initialization is being used, the initialized value of the CRC generator shall be set to the modulo 2^{32} sum of the four 32-bit addends shown in figure 25. The most significant bit of the initialized value of the CRC generator shall correspond to the most significant bit of the sum (see Note below). The channel image ID, control-unit image ID, S_ID, and D_ID are those being used in the exchange on which the data is sent. For a command-data DIB, a data DIB, and a status DIB, the device address is that being used on the exchange on which the data is sent and the CCW number is that of the command IU for which the data is being sent. For a link-control DIB, the device address and CCW number fields are set to zero.

<table>
<tr><td>0</td><td>hex’00’</td><td>CH Image ID</td><td>hex’00’</td><td>CU Image ID</td></tr>
<tr><td>1</td><td>hex’00’</td><td colspan="3">S_ID</td></tr>
<tr><td>2</td><td colspan="3">D_ID</td><td>hex’00’</td></tr>
<tr><td>3</td><td colspan="2">Device Address</td><td colspan="2">CCW Number</td></tr>
<tr><td></td><td>0</td><td colspan="2">15</td><td>31</td></tr>
</table>

Figure 25 – Addends of the Alternative Initialized Value of the CRC Generator

The initialized value of the CRC generator shall be represented by the coefficients of the polynomial “L(x)” in equations (2) and (4), Annex A of FC-FS-4. When optional CRC generation initialization is being used, the bits of the modulo 2^{32} sum of the quantities in figure 25 shall be used as coefficients in the polynomial L(x). The most significant bit (bit 0) of the modulo 2^{32} sum corresponds to the coefficient of X^{31}; lower order bits of the modulo 2^{32}sum correspond to lower-order coefficients in the polynomial L(x), respectively.

NOTE 26 – The preferred implementation is for CRC generation and checking to be provided. This provides end-to-end protection for data at the FC-4 processing layer. Optional alternative CRC generation initialization provides a higher level of protection by enabling the recipient of an IU to detect a larger percentage of possible errors caused by lost frames and possible errors caused by the association of incorrect header information with a block of data.

8.7 Command DIB Structure

8.7.1 Command DIB Overview

A command DIB shall be sent only in a command-mode command IU by the channel. It shall be used to transfer information associated with the current CCW to the control unit. At the beginning of an I/O operation, a command IU containing a command DIB may be used to initiate an operation with a device. When data chaining is performed for a read operation, a command IU containing a command DIB shall be used to update the information held about the current command at the control unit.

Each command DIB shall contain a command header and LRC. The structure of the command IU containing the command DIB is shown in figure 16.

8.7.2 Command Header

8.7.2.1 Command Header Format

The command header shall be the first 12 bytes of either a command DIB or command-data DIB. The command header shall contain both information from the current CCW and information supplied by the channel necessary for the current command to be executed. The command header describes how execution of the I/O operation being initiated is to be performed and may also specify how a subsequent I/O operation is to be performed.

The command header has the format shown in figure 26.

0	CCW Command	CCW Control Flags	CCW Count	
1	Reserved	I/O Priority	Reserved	Command Flags
2	Reserved	IU Count	Data Byte Count	
	0 7	15	23	31

Figure 26 – Command Header

Bytes 0 and 2 of word 1 and byte 0 of word 2 shall be reserved and set to zero by the channel and ignored by a control unit.

8.7.2.2 Command Field

Byte 0 of word 0 of the command header shall contain the command specifying the I/O operation to be executed with a device. The basic operations shall be specified by the following commands: read, read backward, write, control, and sense.

The rightmost bit positions shall indicate the type of operation; the leftmost bit positions shall indicate a modification code which expands the meaning of the basic command that is to be executed. The modifier codes and the commands executed when they are decoded are model dependent.

Table 12 lists the basic commands and the bit settings of the command field. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 12 – Contents of the Command Field

Command	**Bit Position**							
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Reserved	M	M	M	M	0	0	0	0
Sense	M	M	M	M	0	1	0	0
Reserved	M	M	M	M	1	0	0	0
Read Backward	M	M	M	M	1	1	0	0
Write	M	M	M	M	M	M	0	1
Read	M	M	M	M	M	M	1	0
Control	M	M	M	M	M	M	1	1
Explanation: M represents a modifier bit.								

When the command is initiating a device operation, the command field may contain any value that is not reserved. If a reserved command code is received by a device, the device shall respond with unit-check status, and the sense data associated with the unit check shall indicate command reject.

There are some commands (particular combinations of eight bits in the command field) which shall be executed by all devices. Table 13 indicates these required commands.

Table 13 – Required Commands

Command Code (hex)	Function
04	Basic Sense
03	No-operation (no-op)
md	Read configuration data
md	Read node identifier
E4	Sense ID
md	Set interface identifier

Explanation:

md Model-dependent command code. The command code may be obtained by using a sense-ID command.

When a command IU is used to update the count and flags during data chaining (the DU flag of the command DIB header is set to one), the contents of the command field is unpredictable and shall be ignored by the control unit receiving the command IU.

Some commands, when executed, do not result in the transfer of data but either cause the device to respond to the command with status which contains channel end, with or without device end (and without busy status) or cause the device to chain and start execution of the next command for the case where all of the conditions for not transferring status are satisfied. When this occurs, the device operation is referred to as an immediate operation. There are other situations in which an device operation may result in no data being transferred (e.g., when a CCW count of zero has been validly specified). However, the operation shall not be an immediate operation if the CRR CCW control flag is not set for the command and the device responds to the command by sending a command response control IU. Each of the basic operations is described in the following section.

Read: A read command shall initiate execution of a device operation that performs device-to-channel data transfer. The bytes of data within a block shall be provided in the same order as those received by the write command.

Read Backward: A read-backward command shall initiate execution of a device operation in the same manner as the read command, except that bytes of data within a block shall be sent to the channel in an order which is the reverse of that used in writing. The bits within an 8-bit byte shall be in the same order as sent to the device on writing. The control unit may be designed to cause mechanical motion in the device in a direction opposite to that for a read command, or it may be designed to operate the device as it would for a read command.

Unless otherwise noted, any description that applies to read shall also apply to read backward.

Write: A write command shall initiate execution of a device operation that performs channel-to-device data transfer.

Control: A control command shall initiate execution of a device operation that performs channel-to-device data transfer. The device shall interpret the data as control information.

Sense: The sense command is similar to a read command, except that the data shall be obtained from sense indicators rather than from a record source.

8.7.2.3 CCW Control Flag Field

Byte 1 of word 0 shall contain the CCW control flags. Bits 3 and 5-7 shall be reserved and shall be ignored by the control unit receiving the command IU. The value to which bits 3 and 5-7 are set is unpredictable. The format of the CCW control flag field shall be as shown in figure 27.

CD	CC	SLI		CRR			
0	1	2	3	4	5	6	7

Figure 27 – CCW Control Flag Field

Chain Data (CD): The CD flag, bit 0, when set to one, shall specify an intent to perform chaining of data. It shall cause the CCW control flags, CCW count, and command flags designated in the next command IU in which the DU flag and the CH bit are set to ones to be used with the current command. When the CD flag is set to one, the CC flag shall be ignored by the control unit (see 9.2.2.4). If a command IU with the DU flag set to one is followed by a command IU specifying a command, the CD flag shall be set to zero and the CC flag shall be set to one.

When data chaining occurs, a new set of CCW control flags, CCW count, and command flags shall go into effect for the current command.

It is model dependent whether the control unit permits data chaining for a particular command or for a particular device. When the control unit does not support data chaining or if the command is rejected for another reason, the control unit shall reject the command in the first command IU of a data chain (i.e., a command IU with the CD flag set to one and the DU flag set to zero) with unit-check status. If the command is rejected due to data chaining, unit-check status shall be returned in a solicited data IU containing a status DIB sent either in direct response to the command IU or after sending a command response control IU. However, in either case, any data received for the rejected command shall be discarded by the control unit. When data chaining is not permitted, the sense data associated with the unit check shall indicate command reject along with a model-dependent indication that the unit check occurred because data chaining is not permitted for the command.

Chain Command (CC): The CC flag, bit 1, when set to one while the CD flag is set to zero, shall specify an intent to perform chaining of commands. Upon normal completion of the current I/O operation and after recognition of device end at the device, chaining of commands shall cause the command code specified in the command header of the next command IU to be initiated.

In the channel or in the control unit, when the CD flag is set to one, the CC flag shall be ignored.

When command-chaining occurs at the channel, the command, CCW control flags, and CCW count from the current CCW along with the command flags shall be transferred from the channel to the control unit in a command IU; the CH bit shall be set to one in the device-header-flags field; and the

DU flag shall be set to zero in the command-flag field of the command header. Other command flag bits may also be set to one.

In the case where command chaining at the channel is the direct result of having received normal status containing device end, the CH bit set to one and the DU flag set to zero shall indicate acceptance of this status by the channel.

When command chaining occurs at the control unit, the command, CCW control flags, CCW count, and command flags contained in the command header of the next command IU received, with the CH bit set to one and the DU bit set to zero, shall take control.

Suppress Length Indication (SLI): The SLI bit, bit 2, shall control whether or not command chaining is to occur on an incorrect-length condition, that is a condition recognized when the CCW count is not equal to the number of bytes requested or offered by the device. When the SLI bit is set to one and an incorrect-length condition exists for the current command, command chaining, if indicated, shall be permitted. If the SLI bit is set to zero and an incorrect-length condition exists for a command that is executed as a non-immediate operation, command chaining, if indicated, shall not be permitted and ending status with the residual count and appropriate status flag bits shall be transferred to the channel. If the SLI bit is set to zero and an incorrect-length condition exists for an immediate command, command chaining, if indicated, may or may not occur depending on the setting of the COC command flag bit. If the COC command flag bit is set to zero command chaining shall not be permitted and ending status with the residual count and appropriate status flag bits shall be transferred to the channel (see 8.7.2.6).

Command Response Request (CRR): The CRR flag, bit 4, when set to one for a command IU, other than the first command IU of a channel program, shall cause the control unit to indicate that it has started execution of that command by sending a command-response IU.

The CRR flag shall only have meaning when the CH bit in the DH_Flags field is set to a one. The control unit shall ignore the CRR bit in the first command IU of a channel program, but not in subsequent command IU’s that are command-chained or data-chained.

When a control unit starts execution of a command received as a result of command chaining or command retry and for which the CRR bit is set to one, a command-response IU shall be sent to the channel.

When both the CRR CCW control flag and the REX command flag are set to one, a control unit shall send only one command-response IU, independent of the number of times the command is executed.

The channel shall use the CRR flag in order to solicit a command response from the control unit when using the IU pacing function (see 9.2.2.5). The conditions under which the channel uses the CRR flag in this manner and the frequency it is used during the channel program are model dependent. The channel may also periodically set the CRR flag to one in a command IU other than the first command IU of a channel program for purposes of tracking the execution of the channel program. For example, when the channel is executing a CCW with the Program-Controlled Interruption (PCI) flag set to a one, which is not the first command of a channel program, it shall set the CRR flag in the command IU.

8.7.2.4 CCW Count Field

Bytes 2 and 3 of word 0 shall contain the byte count specified in the current CCW.

The CCW count field is a 16-bit field that shall indicate the number of bytes to be transferred between the channel and control unit during execution of the CCW, not including the pad and CRC bytes (see

8.6.4). The field shall be interpreted as a 16-bit unsigned binary integer. The value in the field may range from 0 to 65,535.

8.7.2.5 I/O Priority Field

Byte 1 of word 1, shall contain the control-unit I/O priority number. This number shall form an unsigned binary integer that shall specify the priority level that is applied at the control unit during execution of a CCW channel program. I/O priority provides a means for specifying an end-to-end priority. The usage of I/O priority is beyond the scope of FC-SB-6. The I/O priority value is obtained from the control unit priority field in the operation-request block (ORB). For additional information regarding the ORB, see Annex E.

The specified control-unit I/O priority number may be any value in the range of 1 to 255; and it shall be contained, unchanged, in byte 1 of word 1 for each command DIB or command-data DIB comprising the channel program in execution. The control unit shall not check the I/O priority field in any command DIB or command-data DIB subsequent to the first command or command-data DIB of the channel program. The number 1 shall designate the lowest priority that may be assigned to the I/O-operations at the control unit and the number 255 shall designate the highest priority that may be assigned. The number 0 shall mean that no priority is assigned to the associated I/O operations. The handling of I/O-operations, in this case, shall depend upon the control unit model.

Depending upon the model, fewer than 255 priority levels may be supported by the control unit in which case, the control unit shall round the specified priority to the equivalent level of priority that it supports. For example, if the control unit supported 127 levels of priority, then a specification of 254 would result in an equivalent priority of 127. Also depending upon the model, the control unit may not support the function of I/O priority, in which case, byte 1 of word 1 shall be ignored.

NOTE 27 – A control unit should implement a model-dependent fairness algorithm which ensures that no channel is permanently prevented from accessing a device because it is setting the I/O priority to a lower value relative to the I/O priority used by another channel.

8.7.2.6 Command-Flag Field

Byte 3 of word 1, shall contain the command flags. The channel shall use the command flags to provide the control unit with additional information on how the I/O operation, specified by the contents of the CCW in the command header, is to be executed.

Bits 0-2 shall be reserved, shall be set to zeros by the channel, and shall be ignored by a control unit. The format of the command flag field shall be as in figure 28.

Reserved	DU	COC	SYR	REX	SSS
0 2	3	4	5	6	7

Figure 28 – Command-Flag Field

Data-Chaining Update (DU): The DU flag, bit 3, when set to one, shall indicate that the CC flag, the CD flag, and the count sent in this command IU are associated with a new CCW used during data chaining. The CH bit in the device-header-flags field shall be set to one when the DU flag is set to one.

Continue on Command Immediate (COC): The COC flag, bit 4, when set to one along with the CC flag also set to one shall indicate that suppression of the incorrect length condition is recognized by the channel and command chaining to the next command is permitted at the end of execution of a command immediate operation. When a command is executed as an immediate operation and the CCW count for the command is nonzero, an incorrect length condition shall be recognized and command chaining, if indicated, shall be under control of the COC flag or SLI flag. If the COC flag or the SLI flag is set to one, command chaining, if indicated, shall be permitted. If the COC flag and the SLI flag are both set to zero, command chaining shall not be permitted (see 8.7.2.3).

Synchronize Response (SYR): The SYR flag, bit 5, when set to one, shall indicate to the control unit that the command IU contains a command being sent in response to status received for the immediately preceding command. The channel may set the SYR bit to one only when the CH bit is set to one and the DU bit is set to zero in the command header.

Repeat Execute (REX): The REX flag, bit 6, may be set to one only for an output operation and when the command and all of the data associated with the operation is contained in a single IU. When the REX flag is set to one, it shall indicate that the channel is requesting the control unit to perform the transfer-in-channel function by repetitively executing the command in the command header.

When the channel has fetched a CCW containing a command, all of the conditions for setting the REX bit to one are satisfied, and the next CCW (the one to which chaining occurs) contains a transfer-in-channel command with an address of the previous CCW, it is recommended that the channel set the REX bit to one in the command IU. Transfer-in-channel commands shall not be sent to the control unit.

When the control unit recognizes this indication, re-execution of the current command shall be performed until one of the following conditions occur:

a) execution of the command results in status of other than channel end and device end alone being recognized;

b) execution of the command has been performed 128 times; or

c) execution of the command is terminated by a cancel, selective reset, or system reset.

The REX flag shall be set to one only when the CD, DU, and SSS flag bits in the command header are all set to zero. If a command IU is received with the REX bit set and any of the CD, DU, or SSS flag bits are set to one, a device-level protocol error shall be recognized.

When an execution of the command results in status of channel end, device end, and status modifier alone being recognized, chaining to the next command shall occur at the control unit.

When re-execution of the command is performed 128 times without status other than channel end and device end alone being recognized, re-execution of the command shall be terminated and the status recognized for the last execution of the command shall be sent to the channel.

When execution of the command results in status of other than channel end and device end alone or channel end, device end, and status modifier being recognized, re-execution of the command shall be terminated at the control unit and status shall be sent to the channel (see 8.10).

When the REX flag is set to zero, re-execution of the command is not requested.

If for certain commands the control unit does not support the repeat execute function, then it may always reject this command with unit-check status. If for other commands it does support the repeat execute function, it shall accept and execute the command with the REX bit set to one as required by the REX bit. If for all commands it does not support the repeat execute function, it may reject all commands with the REX bit set to one with unit-check status. Whenever a command is rejected with unit-check status because the REX bit is set to one, model-dependent sense data shall be provided.

Synchronize Send Status (SSS): The SSS flag, bit 7, when set to one, shall indicate that the channel has recognized a condition requiring synchronization with I/O execution at the control unit and is requesting that status be generated and presented at the completion of the I/O operation for this command IU. When the I/O operation has been concluded at the device, a status DIB containing device-end status is sent. When channel-end status is not accompanied by device-end status, a status DIB containing channel-end status is sent, and the control unit subsequently sends a status DIB containing device-end status when the I/O operation has been concluded at the device.

When conditions at the channel require synchronization with execution of a command at the control unit, the SSS flag bit shall be set to one in the command IU. If data chaining is indicated for the command, the SSS flag bit shall be set to one in the command IU sent for the last CCW providing data chaining.

When a command IU with the SSS flag bit is sent, command chaining at the channel shall be temporarily suspended. When chaining is suspended at the channel, execution of the channel program shall temporarily end with the execution of the last CCW for the current command, and until initiative to perform command chaining is provided by the receipt of status that allows chaining, command IUs and data IUs shall no longer be sent.

When a control unit completes the execution of a command for which the SSS flag is set, status shall be generated, and sent to the channel. If a command IU is received with both the SSS flag and REX flag set to one, a device-level protocol error shall be recognized.

8.8 Command-Data DIB Structure

A command-data DIB shall be sent only in a command-mode command IU by the channel. It shall be used to transfer both information associated with the current CCW for write commands with a nonzero CCW count and all, some portion or none of the data associated with the command. When data chaining is performed for a write operation, a command-data DIB shall be used to update the information held about the current command at the control unit.

A command-data DIB shall contain a command header, LRC, and a DIB-data field (see figure 16). The format of the command header shall be the same as that of a command DIB. The DIB data field shall be a maximum of 8160 bytes. If the CCW count for the command is 8156 bytes or less, all of the data for the command shall be sent along with the CRC field in the DIB-data field. If the CCW count for the command is greater than 8156 bytes, up to the first 8160 bytes shall be sent in the DIB-data field, and the CRC field and remaining data, if present, shall be sent in one or more subsequent data IUs. If the CCW count for the command is greater than 8156 bytes, and less than 8160 bytes, pad bytes shall be added at the end of the command-data DIB, and the CRC field shall be sent in a subsequent data IU.

The DIB data byte count, bytes 2 and 3 of word 2, of the command header shall specify the number of data bytes contained in the DIB-data field (see 8.6.2.3).

8.9 Data DIB Structure

8.9.1 Data DIB Overview

A data DIB shall be sent only in a solicited command-mode data IU by either the channel or control unit.

For a write operation, the channel shall use one or more data IUs containing a data DIB to transfer any remaining portion of data not sent in the command-data DIB. When the quantity of data to be sent for a write operation exceeds the maximum number of bytes that is allowed to be sent in an IU, the channel shall send a command IU containing a command-data DIB for the command and the maximum number of bytes allowed; followed by one or more data IUs until either the quantity of data specified by the CCW count is satisfied or an early end condition is recognized.

For a read operation, the control unit shall use one or more data IUs containing a data DIB to transfer to the channel the data for the read command. When the quantity of data to be sent requires more than one data IU, the control unit shall send a data IU followed by one or more additional data IUs until one of the following occurs: 1) the quantity of data specified by the CCW count is satisfied, 2) an early end condition is recognized, or 3) a condition requiring status to be sent is recognized.

All data IUs contain the maximum number of bytes, 8160, except the last data IU sent for a CCW which may have less than the maximum.

When the data DIB containing the last data byte for the current CCW contains more than 8156 bytes of data, the CRC field shall not be included in the data DIB and shall be sent in a subsequent data DIB by itself. The data DIB containing greater than 8156 bytes of CCW data shall have pad bytes added, if necessary, and the CRC value sent in the subsequent data IU (if generated) shall cover both the CCW data and the pad bytes. In the last data IU, containing only the 4-byte CRC field, the byte count shall be set to zero.

The CCW number and token provided in the IU header of each data IU sent shall be the same as that provided in the command IU.

8.9.2 Data Header

The data header is the first 12 bytes of a data DIB. The data header shall have the format shown in figure 29.

0	Reserved		
1	Reserved		
2	Reserved	IU Count	DIB Data Byte Count
	0 7	15	31

Figure 29 – Data Header

Word 0, word 1, and byte 0 of word 2 shall be reserved and set to zero by the sender and ignored by the recipient.

8.10 Status DIB

8.10.1 Status DIB Processing

A status DIB shall be sent in either a solicited command-mode data IU or unsolicited command-mode data IU by a control unit. It shall be used to transfer status, with or without supplemental status, to the channel (see figure 16). A solicited data IU containing a status DIB may be sent in direct response to a command IU, or to certain initiation IUs when the request-status, selective-reset, or cancel control function is specified.

Additionally, a solicited data IU containing a status DIB may be sent during the data transfer portion of an exchange.

An unsolicited data IU containing a status DIB shall be used to present asynchronous status or any status that initiates a connection and is not sent in direct response to an IU received from the channel.

After sending a data IU containing a status DIB, the control unit shall discard all command IUs with the SYR flag set to zero and data IUs associated with that device address and logical path, either queued at the control unit, or received from the channel; discarding continues until a command IU with the SYR flag set to one is recognized by the control unit for the device and logical path. Acceptance of the status by the channel may be indicated explicitly when a status-accepted IU is received or may occur implicitly when a command IU with the SYR bit in the command header set to one, is received.

When any of the following conditions exist, status shall be sent to the channel at the completion of the current I/O operation.

a) The SSS command flag bit is set to one.

b) An early end indication is received.

c) The CC flag bit is set to zero.

d) The first command of the channel program is executed as an immediate operation.

e) The CC flag bit is set to one and the SLI flag bit is set to zero and an incorrect length condition is detected for a command that is executed as a non-immediate operation.

f) The CC flag bit is set to one, both the SLI flag bit and COC command flag bit are set to zero, the CCW count is nonzero, and the command is executed as an immediate operation.

g) The REX bit is set to a zero and status other than channel end (with or without device end) is recognized.

h) The REX bit is set to a one and status other than channel end and device end, or channel end, device end and status modifier is recognized.

i) The REX bit is set to one, and the command has been executed 128 times.

j) Status of channel end (without device end) is recognized, and conditions at the control unit require a disconnection from the channel to suspend the transferring of command IUs and data IUs.

If none of the above conditions exist, status shall not be sent and chaining, if indicated at the control unit, shall be permitted (see Annex D). Additionally, status shall not be sent if during an existing exchange any previously sent status has not yet been accepted by the channel.

For a status DIB, the AS bit may be set to either zero or one depending on whether the status is associated with the control unit or a device. If the status contains one of the following combinations, the status is associated with the control unit, and the AS bit shall be set to zero.

a) Control Unit Busy. If a control unit receives an IU that initiates an exchange and a busy condition exists which prevents the acceptance of any FC-SB-6 IU with the exception of a control IU indicating system reset, then the control unit shall send a solicited data IU with a status DIB with only the status-modifier and busy status bits set to ones and the AS bit set to zero. This status is referred to as control-unit-busy status (see 9.4.10).

b) Control-Unit End Alone. If a control unit does not recognize a stack status sent in response to a status DIB containing control-unit-busy status, the control unit shall later send a data IU containing a status DIB with the control-unit-end status bit set to one. When all other status bits are set to zeros, the AS bit is set to zero, and the status is referred to as control-unit end alone.

c) Control-Unit End with Busy. When control-unit-end status is pending and the control unit recognizes a valid command IU that initiates an exchange pair, the control unit may return a solicited data IU with a status DIB with only the control-unit-end and busy status bits set to ones and the AS bit set to zero. This status combination is referred to as control-unit end with busy.

d) Control-Unit End with Control Unit Busy. If a control unit receives an IU that initiates an exchange pair and the IU is not a control IU indicating system reset, the control unit may return a solicited data IU with a status DIB and with only the status-modifier, control-unit-end, and busy status bits set to ones, and the AS bit set to zero. This status combination is referred to as control-unit end with control unit busy. When the channel recognizes this combination of status bits, shall consider the status to be an indication that the control unit is not busy.

In all other cases, the status shall be associated with a device address, and the AS bit shall be set to one; if the AS bit is set to zero with any status combination not listed above, the channel shall detect a device-level protocol error.

A data IU with the AS bit set to zero containing a status DIB shall be sent only if the status is one of the preceding combinations and meets one of the following conditions.

a) The status is control-unit end alone and is to be sent in an unsolicited data IU.

b) The status is in response to a valid request-status IU that initiated a connection, and the status byte is not control-unit end with busy.

c) The status is in response to a valid command IU that initiated a connection, and the status is not control-unit end alone.

d) The status is in response to a valid IU other than a system-reset IU, that initiated a connection, and the status is control unit busy.

The channel shall detect a device-level protocol error if a data IU containing a status DIB is received that has the AS bit set to zero and does not meet one of the preceding conditions.

NOTE 28 – Control units and devices should provide interlocks so that status is not lost, hidden, or included with other status when the result would cause the program or channel to misinterpret the original meaning and intent of the status.

NOTE 29 – The protocol prohibits the sending of status except under the conditions listed in order to minimize the performance impacts caused when status is presented.

8.10.2 Status DIB Structure

The format of the data IU for a status DIB is shown in figure 16. A status DIB shall consist of a 12-byte status header, 4-byte LRC, and when supplemental status is included, from 1 to 32 bytes of sense data (and, if necessary, pad bytes) and the 4-byte CRC field. When supplemental status is not included in the status DIB, the status DIB shall consist only of the status header and the 4-byte LRC.

8.10.3 Status Header

8.10.3.1 Status Header Format

The status header is the first 12 bytes of a status DIB. The status header shall have the format shown in figure 30:

0	Status Flags	Status	Status Parameters
1	Queue-Time Parameter		Defer-Time Parameter
2	Reserved	IU Count	Supplemental Status Byte Count
	0 7	15	31

Figure 30 – Status Header

Byte 0 of word 2 shall be reserved and set to zero by a control unit and ignored by the channel.

8.10.3.2 Status-Flags Field

Byte 0 of word 0 of the status header shall contain the status flags. The status-flag field shall be used to provide additional information to the channel concerning the conditions that were present at the control unit when status was generated and conditions that pertain to the status DIB. These flags shall assist the channel in determining how to handle the status DIB and what status, if any, to report. The status-flag field shall have the format shown in figure 31:

FFC			CI	0	CR	LRI	RV
0	1	2	3	4	5	6	7

Figure 31 – Status-Flag Field

Flag-Field Code (FFC): The FFC, bits 0-2, is a three-bit encoded field that either in conjunction with or independent of the other status flag bits shall further describe the status information contained in the status byte, the status-parameter field, and the queueing-information fields. The flag-field code assignments shall be:

000 No Function. The status byte, status-parameter field, and queueing information fields are not affected by this FFC code.

001 Queueing Information Valid. The queue-time-parameter field contains control-unit queueing information. This code is set by the control unit and checked by the channel.

010 Resetting Event. A resetting-event condition exists for the logical path and the device for which unit-check status is being presented. This code shall be permitted only when unit-check status is presented for a resetting-event condition and then only as initial status in response to a command IU for the first command of the channel program; if this code is indicated at any other time, the channel shall detect a device-level protocol error (see 9.6).

011-111Reserved.

Channel Initiated (CI): The CI bit, bit 3, when set to one, shall indicate that this solicited data IU containing a status DIB is in direct response to a control IU indicating selective reset with either a request for retry or a request for unit check (see 9.5.3.2 and 9.5.3.3). If the CI bit is set to zero, this data IU containing a status DIB is not in direct response to a control IU indicating selective reset with a request for retry or a request for unit check.

Reserved: Bit 4 shall be reserved.

Command Retry (CR): The CR bit, bit 5, when set to one, shall be used to request command retry if the status byte also contains retry status. If the CR bit is set to zero or the status byte does not contain retry status, command retry is not being requested. The status byte shall contain retry status if it contains unit check and status modifier together with 1) channel end alone (meaning the control unit or device is not yet ready to retry the command), or 2) channel end and device end (meaning the control unit and device are prepared for immediate command retry). All other status combinations shall not be considered retry status; if the CR bit is set to one with any other status combination, a device-level protocol error shall be detected.

Long Record/Immediate (LRI): The LRI bit, bit 6, when set to one, shall indicate that either a long-record condition was detected by the control unit or that the status being presented with a residual count equal to the CCW count is associated with a command that was executed as an immediate operation.

When the control unit detects a long-record condition (that is, additional data would have been sent to the channel or written to the device beyond the data provided for the current command), the LRI and RV bits shall be set to one and the residual count shall be set to zero. When data chaining is not used, the amount of data provided for the current command is equal to the count provided in the CCW, except when early end is signalled. When data chaining is used, the amount of data provided for the current command is equal to the sum of the counts of all CCWs used for the current command.

When the control unit executes a command, that is not the first command of a channel program, as an immediate operation and the CCW count is greater than zero, the LRI and RV bits shall be set to one and the residual count shall be set to the value of the CCW count. When the control unit executes the first command of a channel program as an immediate operation, and the CCW count is greater than zero, the LRI bit is meaningless, may be set to zero or one, and shall not be checked by the channel.

The LRI bit may be set to one by the control unit only when the channel-end status bit is set to one. If the channel-end status bit is set to zero in a status DIB, the control unit shall set the LRI bit to zero, and the LRI bit shall be ignored by the channel.

See table 14 for a summary of the usage of the LRI bit.

Residual-Count Valid (RV): The RV bit, bit 7, when set to one, shall indicate that the status-parameter field contains the residual count. For write commands, the residual count is equal to the difference between the CCW count for the write command and the number of bytes actually written to the device. For read commands, the residual count is the difference between the CCW count and the number of bytes actually read from the device and transferred to the channel. The RV bit may be set to one by the control unit only when the channel-end status bit is set to one. If the channel-end status bit is set to zero, the control unit shall set the RV bit to zero and the RV bit is ignored by the channel.

When status is sent in response to the first command of a channel program that is executed as an immediate operation, the RV bit shall be set to zero.

When status with channel end is sent in response to a command other than the first command of a channel program that is executed as an immediate operation, the RV bit shall be set to one unless an abnormal condition exists that precludes calculation or transfer of the residual count, in which case the RV bit shall be set to zero and unit check shall be indicated in the status byte.

When the LRI bit is meaningful and set to one, the RV bit shall be set to one (see 8.10.3.2).

When the RV bit is set to zero, both the E and LRI bits shall not be meaningful, and the status-parameter field shall not contain a residual count (see 8.10.3.4).

Table 14 summarizes the use of the RV and LRI status flags and the channel-end status bit. In addition to these bits, the E bit in the device-header-flags field shall be set to one if during a write command an early end condition was indicated and the data sent by the channel is equal to the record length. In all other cases, the E bit is set to a zero for a status DIB.

Table 14 – LRI and RV Bit Usage

Condition	CH END	LRI	RV	Status Parameter Field
CCW count = data read or written	1	0	1	Residual Count = 0
CCW count < # bytes in record	1	1	1	(Note 1)
READ Command: CCW count > # bytes read and transferred	1	0	1	CCW count minus # bytes read & transferred
WRITE Command: CCW count > # bytes written	1	Note 2	1	CCW count minus # bytes written
Non-Immediate Command: CCW count > 0	1	Note 2	1	CCW Count (no data transferred)
Immediate Command (not first) CCW count > 0	1	1	1	CCW count
Immediate Command (first) CCW count > 0	1	X	0	IU Pacing parameter (Note 3)

Notes:
1. If the channel indicates early end, the residual count = CCW count minus # bytes written, otherwise the status parameter field is set to zero.
2. If the channel indicates early end, the LRI bit is set to one if the CCW count < # bytes in record, otherwise the LRI bit is set to zero.
3. See 8.10.3.4 for additional information.
4. X = don’t care.

8.10.3.3 Status Byte

Byte 1 of word 0 shall contain the status byte. The status byte indicates device and control unit status. The status byte shall have the format shown in table 15. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 15 – Status Byte

Status Bit Position	Description
0	Attention
1	Status modifier
2	Control-unit end
3	Busy
4	Channel end
5	Device end
6	Unit check
7	Unit exception

A brief summary of each of these status bits follows.

If the channel receives a status DIB with the status byte set to zero, a device-level protocol error shall be detected. The status DIB shall contain a status byte with a combination of status bits set to ones which is appropriate for the conditions existing when the status is presented. If the combination of status bits is not appropriate for the existing conditions, the channel may detect a device-level protocol error.

The channel shall indicate acceptance of the status by means of either a status-accepted control IU, or, when retry or chaining is performed and the status includes device end, by means of a command IU having the CH bit set to one.

See 8.10 for the situations in which the following status conditions are sent to the channel.

Attention: The attention condition shall be generated when some asynchronous condition occurs in the control unit or device. The condition may be accompanied by other status. Attention shall not be associated with the initiation, execution, or termination of any I/O operation.

Status Modifier: Status modifier shall be generated by the device when the device is not able to provide its current status in response to a status request by the channel subsystem, when the control unit is busy, when the normal order of commands has to be modified, or when command retry is to be initiated.

Control-Unit End: Only control units that indicate a control-unit-busy condition shall indicate a control-unit-end condition. Control-unit-end status shall be returned from the control unit to the channel after the channel accepts control-unit-busy status from the control unit. Control-unit-end status shall be returned after the control-unit-busy condition no longer exists. This is sometimes referred to as no-longer-busy status for the control unit. Only one control-unit-end indication shall be returned on a logical path, regardless of the number of times the channel accepted control-unit-busy status on that logical path during the busy period.

The control unit shall not associate pending control-unit-end status with any device address. A selective reset shall not reset pending control-unit-end status. If the channel stacks or does not accept a status byte that contains the control-unit-end status bit, the control-unit-end status shall not be held along with the status byte, and it shall remain pending and unstacked at the control unit. Control-unit-end status may be withdrawn by the control unit if the control unit becomes busy again before the status is accepted by the channel; in such a case, the control unit shall return control-unit-end status later, after the control-unit-busy condition no longer exists.

When control-unit end is included with other status bits set, other than those status combination required to have the AS bit set to zero, the AS bit shall be set to one, and the device address used shall be the device address for which the other status is being sent (see 8.10). A pending control-unit end shall not necessarily preclude initiation of new operations. Whether the control unit allows initiation of new operations is at the option of the control unit.

Control-unit end shall not necessarily cause command chaining to be suppressed. Control-unit end shall not cause command chaining to be suppressed when presented with the AS bit set to zero as described in 8.10.

When control-unit-end status is presented on a logical path along with status modifier and busy, with no other status bits set to one, the combination should be interpreted as control-unit-end status. When this status is accepted, the control unit shall no longer owe a control-unit-end status indication on that logical path. If the control unit was interrogated while it was in the busy state, and then system reset is recognized by the control unit before control-unit-end status is accepted by the channel, the control unit shall not owe control-unit-end status.

A control unit shall only present control-unit-end status when a no-longer-busy condition is owed. However, the channel shall not detect an error if control-unit-end status is received when no control-unit-busy condition was indicated (that is, a no-longer-busy condition was not owed).

NOTE 30 – Presentation of the control-unit-end status bit without any other status bits is the preferred implementation.

Busy: The busy indication shall occur only when conditions existing at the device or control unit preclude execution of the intended I/O operation because of one of the following four situations.

a) A previous I/O operation or chain of I/O operations is being initiated or being executed.

b) Stacked or pending status conditions exist, and the pending status conditions are returned in response to a command IU, busy is appended to the status returned.

c) The control unit is shared by channels or devices, or a device is shared by control units, and the shared facility is not available.

d) A self-initiated function (e.g., microdiagnostics or data movement internal to the device) is being performed.

Status conditions for the addressed device, if any, shall accompany the busy indication.

Channel End: Channel end shall indicate that the portion of an I/O operation involving transfer of data or control information between the channel and a control unit has been completed. Acceptance of channel-end status shall indicate the completion of the channel portion of the I/O operation.

Each I/O operation initiated at the device shall cause one and only one channel end condition to be recognized at the control unit, for which channel-end status may or may not be generated and presented to the channel.

Device End: Device end shall be recognized at the control unit (1) when the device portion of an I/O operation is completed, (2) when the device, having previously responded busy, transitions from the busy to the not-busy state, (3) when the device changes from the not-ready to the ready state, and (4) when the control unit or device recognizes an asynchronous condition.

Unit Check: Unit check shall indicate that the device or control unit has detected an unusual condition that is detailed by sense information (see 8.10.4.2). The occurrence of unit check may indicate that a programming error or an equipment error has been detected, that the not-ready state of the device has affected the execution of the command, or that an exception condition other than the one identified by unit exception has occurred. The unit-check bit provides a summary indication of the conditions identified by sense data.

Unit Exception: Unit exception shall mean that the device detected an unusual condition that needs to be reported to the program, such as recognition of a tape mark, and does not necessarily indicate an error.

8.10.3.4 Status-Parameter Field

The status-parameter field, bytes 2 and 3 of word 0 of the status header, is a 16-bit field that may contain a residual count or IU pacing parameter. If the conditions are such that neither the residual count nor IU pacing parameter is to be presented, the control unit shall set the status-parameter field to zero, and the channel receiving the status DIB shall ignore the status-parameter field.

Residual Count: The residual count is a 16-bit unsigned binary number that represents the difference between the CCW count for a command and the quantity of data actually transferred either from or to the device for that command.

For write commands, the residual count shall be equal to the difference between the CCW count of the current CCW for the command and the number of bytes actually written to the device. For read commands, the residual count shall be the difference between the CCW count for the current CCW for the command, and the number of bytes actually read from the device and transferred to the channel. The residual count shall be meaningful only when the residual-count-valid (RV) bit is one.

Persistent IU Pacing Parameter Valid: When persistent IU pacing is enabled and the residual-count-valid (RV) bit is set to zero, the persistent IU pacing parameter valid bit is provided in the status parameter field bit 0. When set to one, this bit indicates that the IU pacing parameter is valid. When set to zero, this bit indicates that the IU pacing parameter is not valid, is ignored by the channel and shall not change the persistent pacing credit at the channel for the logical path. This bit is only meaningful when persistent IU pacing is enabled.

IU Pacing Parameter: The IU pacing parameter is an eight bit value that is carried in the least-significant byte of the status-parameter field.

The use of the IU pacing parameter in the status parameter field depends on whether or not persistent IU pacing is enabled. When persistent IU pacing is not enabled, the IU pacing parameter is provided in the status-parameter field when status is presented for the first command of a channel program that is executed as an immediate operation or when presenting device end in order to reconnect when the chaining condition is set. When persistent IU pacing is enabled, the residual-count-valid (RV) bit is set to zero and the persistent IU pacing parameter valid bit is set to one, the IU pacing parameter shall be provided in the status parameter field.

The IU pacing parameter shall be sent by the control unit to indicate to the channel the maximum number of IUs a channel may send on a given outbound exchange until it receives a command response IU, which was sent because the CRR bit was set to one, on the existing inbound exchange. An IU pacing parameter of zero indicates that the control unit wishes to leave the default value of the IU pacing credit unchanged. See 9.2.2.5 for additional information.

8.10.3.5 Queue-Time Parameter

When control-unit queueing is provided, bytes 0 and 1 of word 1 contain the queue-time parameter (QTP) (see figure 32). Queue time represents the time the I/O operation is queued at the control unit. When control unit queueing is provided, I/O operations specified by the channel program shall be executed up to a point where the I/O resources are unavailable to execute an I/O operation in that channel program. (Such resources would typically be in use executing I/O operations specified in other channel programs). Depending upon the control unit design, queueing of a command may occur after receipt of the first command or after receipt of a subsequent command within the channel program. I/O operations executed up to and including the command to be queued may involve data transfer as a normal course of I/O operation execution. In all cases, queueing of a command for the channel program shall not affect ongoing I/O operations or cause the sequential order of I/O operation execution to be altered. Queueing shall occur only once within execution of the channel program and may occur only between execution of successive I/O operations; however, the point in execution chosen and the length of the queueing time are model dependent. When a command is queued, it shall remain queued at the control unit until either the needed resources become available or when a system reset or selective reset has been executed.

When queueing occurs, the control unit shall either perform a disconnection from the channel by presenting status containing channel end without device end (this status may or may not include command retry status), or temporarily delay, to a later time, further channel program execution without causing a disconnection from the channel. In the former case, execution of the command shall be completed at the channel when the control unit performs the disconnection by presenting channel-end status. When the unavailable resources become available, the queue interval shall be ended when the command is dequeued. The specified I/O operation shall be executed and shall be completed at the device. Connection with the channel occurs and device-end status shall be presented, along with the queue-time parameter. The control unit shall retain the queue-time parameter until it receives acknowledgment that the status transfer was accepted. At this time command chaining, if specified, may occur which allows execution of the channel program to continue. In the latter case, when disconnection from the channel does not occur, the queueing interval shall be the same duration as would have been accumulated had disconnection and subsequent reconnection to the channel occurred. In this situation, the accumulated queueing information shall be held pending at the control unit until either execution of the channel program has completed at the device (device-end status is presented) or a later disconnection and reconnection has occurred and device-end status is presented. In all cases, the control unit shall retain the queueing information until it regards the status as accepted. When this has occurred, all previously stored or accumulated values shall be zeroed at the control unit.

The contents of the queue-time parameter field shall be meaningful only when the flag-field code is set to ‘001’b and the device-end status bit is set to one in the accompanying status byte.

When the queue-time parameter field contains valid information, it shall be obtained in the following manner with the format specified in figure 32.

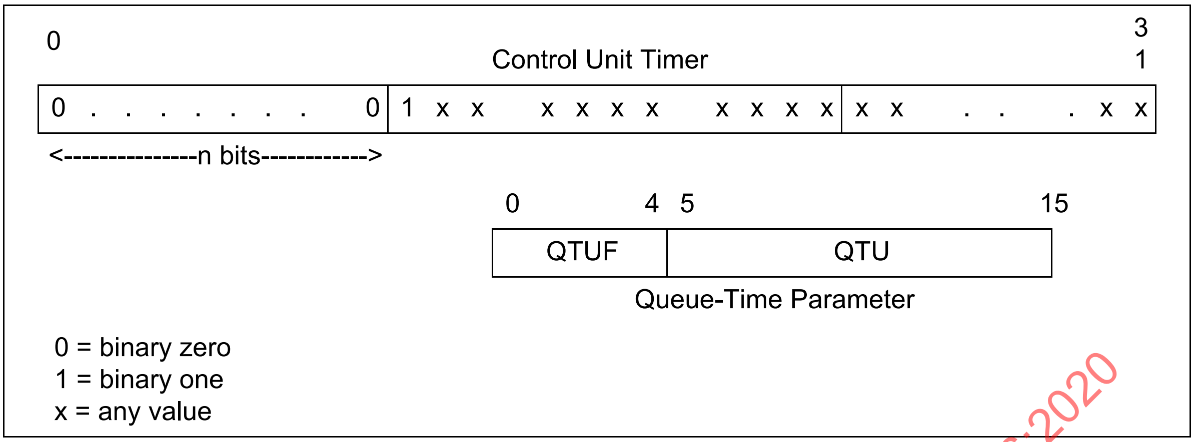


Figure 32 – Queue-Time Parameter (QTP) Format

The queue-time parameter shall be calculated by the control unit using a 4-byte timer having a 1 microsecond resolution. The value accumulated by the 4-byte control unit timer shall specify the queue-time unit to be stored in the queue-time parameter field.

Queue Time Unit Factor (QTUF): Bits 0-4, when non-zero, shall specify the number of bit positions the 11-bit queue-time unit shall be shifted left for interpretation by the channel. If a shift is required, the QTUF shall be calculated by subtracting the value of the highest bit position, which was set to one, of the queue-time unit from 21 (this applies only in those cases where the bit position is equal to or less than 21). If no adjustment is required, the QTUF value shall be ‘00000’b.

Queueing Time Unit (QTU): Bits 5-15 shall contain the 11-bit queueing time unit accumulated by the 4-byte control unit timer. The bit significance of the time unit shall be specified by the QTUF.

Table 16 describes the relationship between the values of QTUF and the bit-significance of QTU.

Table 16 – QTF/QTU Relationship

When QTUF is:	Bit 15 of QTP represents:
00000	1 microsecond
00001	2 microseconds
00010	4 microseconds
00011	8 microseconds
.	.
.	.
.	.
10101	2,097,152 microseconds

In the example given in figure 33, the control unit timer has accumulated a queue time of 75,000 microseconds (hex’124F8’) and calculated the queue-time parameter. When received by the channel, the QTUF indicates a shift of the most significant bit of the QTU to bit position 15 of the channel 4-byte-time value, and bit 15 of the QTU represents a value of 64 microseconds. The low-order 6-bit positions are set to zeros. See 3.3.3 for a definition of the bit-numbering convention used in the figure.

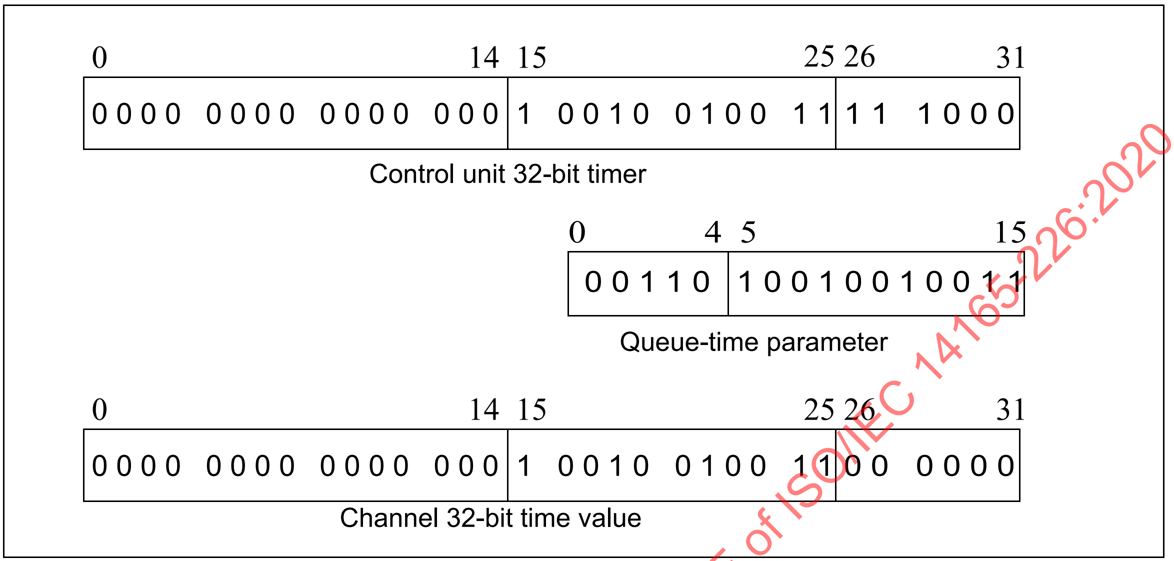


Figure 33 – Queue-Time Parameter Example

A queue time of zero shall be indicated by setting the QTUF to ‘00000’b and the queueing-time unit to ‘00000000000’b.

A control unit that performs queueing shall have a queue-timing facility which is capable of measuring a time interval of up to its expected maximum queueing time, not exceeding 4,294.967295 seconds.

When a control unit converts from a 4-byte timer value to a queue-time unit, if any of the bits of the 4-byte timer value are lost while shifting, the least significant bits shall be discarded (truncated).

Whenever the queue-time-unit factor is value ‘00001’b - ‘11111’b and the queue-time unit is zero, the resultant value of the queue-time parameter (e.g., the values hex’0800, hex’1000’, hex’4000, hex’8000, hex’C000’, etc). shall be not meaningful. This provides for usage of these values to indicate unique conditions recognized by the control unit. In particular, the value hex’8000’ (the QTUF is set to ‘10000’b and the QTU is set to ‘00000000000’b), and the queueing-information-valid code in the flag-field code shall indicate the control unit is not capable of presenting meaningful queue-time as a result of either a queue-timing-facility failure or an overflow condition.

If a queue-timing facility failure has occurred, the control unit shall send a status DIB with unit check status and sense information describing the failure to allow device-dependent program recovery. If a queue-time overflow condition has been recognized, unit check status shall not be presented to the channel. When a queue-timing facility failure has been recognized and after unit-check status has been presented and accepted, the control unit may either:

a) disable its queue-timing facility and process subsequent I/O operations without queueing until this failure condition is corrected; or

b) continue queueing the subsequent I/O operations and present an indication of meaningless queue-time by setting the flag-field code in the status flag field of the status header to ‘001’b and setting the queue-time parameter to the value hex’8000’.

(In the latter case, unit-check status shall not be presented after the initial indication of queueing information not meaningful).

NOTE 31 – Accumulation of excessive defer time when a connection exists may result in the recognition of an FC-SB-6 exchange error. To minimize the probability of this occurrence, the control unit should keep its defer time interval at a minimum by disconnecting from the link.

8.10.3.6 Defer-Time Parameter

When the defer-time function is provided by the control unit, bytes 2 and 3 of word 1 contain the defer-time parameter (DTP) (see figure 34). Device-defer time shall be accumulated by the control unit whenever ongoing channel program execution needs to be temporarily delayed because a device-dependent action needs to be performed. This condition typically occurs whenever the control unit requires the device to access the physical medium for the purpose of reading or writing data. However, because of the device-dependent nature of the action, other conditions may also be recognized by the control unit creating the need for a temporary delay of channel program execution. In all cases, device-defer time shall continue to be accumulated with each successive temporary delay until such time when either the control unit forces a disconnection from the channel or execution of the channel program has been completed. If the control unit disconnects from the channel, the previously accumulated defer-time value shall be presented as the device-defer-time parameter along with status containing channel-end without device end (this status may or may not include command retry status). When the temporary delay of channel program execution is no longer required, the connection to the channel shall be reestablished and device-end status shall be accepted from the control unit. Device-defer time shall not be accumulated during the time interval of a disconnection from the channel. If no disconnection from the channel occurs during execution of the channel program, the previously accumulated device-defer time shall be presented as the device-defer-time parameter along with status of channel end, signaling completion of channel program execution at the channel (status of device end may also be included with the channel-end status). In all cases, the control unit shall retain the defer time information until it regards the status as accepted. When this has occurred, all previously-stored or accumulated defer-time values shall be zeroed at the control unit and accumulation, if any, begins anew. The defer-time parameter shall be obtained in the following manner with the format specified in figure 34.

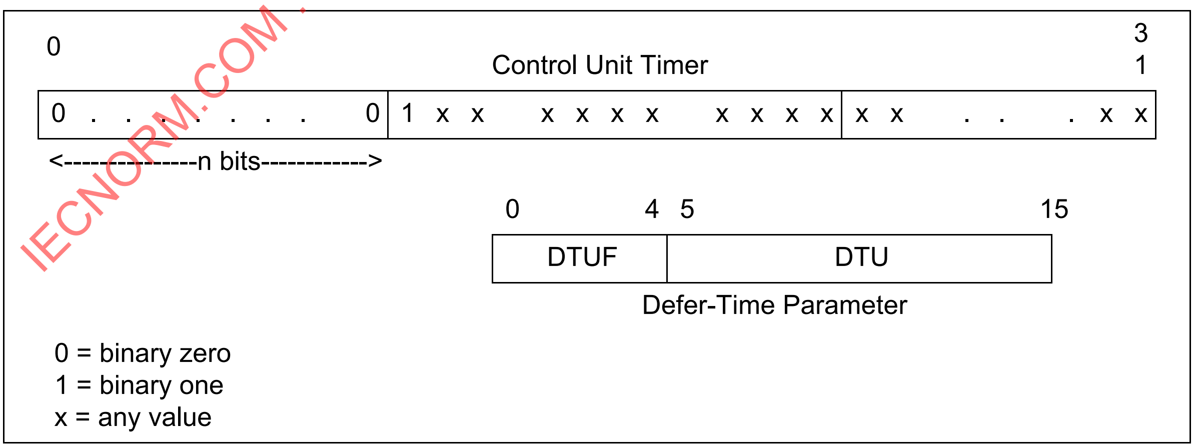


Figure 34 – Defer Time Parameter Format

The defer-time parameter shall be calculated by the control unit using a 4-byte timer having a 1-microsecond resolution. The value accumulated by the 4-byte control unit timer shall specify the defer-time unit to be stored in the defer-time parameter field. When the channel accepts the accompanying status, it uses the control information, and is able to re-establish the 4-byte time value (with low-order truncation).

Defer Time Unit Factor (DTUF): Bits 0-4, when non-zero, shall specify the number of bit positions the 11-bit defer time unit shall be shifted. If a shift was required, the DTUF shall be calculated by subtracting the value of the highest bit-position (which was set to one) of the queue-time unit from 21 (this shall apply only in those cases where the bit position is equal to or less than 21). If no adjustment was required, the DTUF value shall be ‘00000’b.

Defer Time Unit (DTU): Bits 5-15 shall contain the 11-bit defer-time unit accumulated by the control unit timer. The bit-significance of the time unit shall be specified by the DTUF.

Table 17 describes the relationship between the values of DTUF and the bit-significance of DTU.

Table 17 – DTF/DTU Relationship

When DTUF is:	Bit 15 of DTP represents:
00000	1 microsecond
00001	2 microseconds
00010	4 microseconds
00011	8 microseconds
.	.
.	.
.	.
10101	2,097,152 microseconds

In the example given in figure 35, the control unit timer has accumulated a defer time of 9,464 microseconds (hex’24F8’) and calculated the defer-time parameter. When received by the channel, a shift of the most significant bit of the DTU to bit-position 18 of the channel 4-byte-time-value, and bit 15 of the DTP represents a value of 8 microseconds. The low-order 3-bit positions are set to zeros. See 3.3.3 for a definition of the bit-numbering convention used in the figure.

0								8								16		18										28			31
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	1	1	1	0	0	0

Control unit 32-bit timer

0				4	5										15
0	0	0	1	1	1	0	0	1	0	0	1	1	1	1	1

Defer-time parameter

0								8								16		18										28			31
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	1	1	1	0	0	0

Channel 32-bit time value

Figure 35 – Defer-Time Parameter Example

A control unit that performs the defer function shall have a defer-timing facility which is capable of measuring a time interval of up to its expected maximum defer time, not exceeding 4,294.967295 seconds.

When a control unit converts from a 4-byte timer value to a defer-time unit value, if any of the bits of the 4-byte timer value are lost while shifting, the least significant bits shall be discarded (truncated).

Whenever the defer-time-unit factor is value ‘00000’b and the defer-time unit is ‘00000000000’b, either no accumulation shall have occurred, the defer-time function is not supported, a failure of the defer-timing facility shall have been recognized, or an overflow condition shall have been recognized.

If a defer-timing facility failure has occurred, the control unit shall send a status DIB with unit check status and sense information describing the failure to allow device-dependent program recovery. If a defer-timing overflow condition has been recognized, unit check status shall not be presented to the channel. When a defer-timing facility failure has been recognized and after unit-check status has been presented and accepted, the control unit shall continue with execution of subsequent channel programs as before; however, it shall present an indication of meaningless defer-time values by setting the DTUF and DTU fields to zeros (in this case, unit-check status shall not be presented after the initial indication of defer-timing information not meaningful).

NOTE 32 – The preferred implementation for control units that perform the defer-timing function and have detected a defer-timing-facility-failure condition is to continue with channel program execution, when appropriate, and present the indication of meaningless defer time as described previously.

NOTE 33 – Accumulation of excessive defer time when a connection exists may result in the recognition of an FC-SB-6 exchange error. To minimize the probability of this occurrence, the control unit should keep its defer time interval at a minimum by disconnecting from the link.

8.10.4 Supplemental Status Field

8.10.4.1 Supplemental Status Field Overview

Supplemental status, when present, shall provide additional information concerning conditions at the control unit or device for which status information is being provided. Supplemental status shall exist only for unit-check status. When supplemental status is included in a status DIB, it shall be sent in the supplemental-status field. When supplemental status is not included in the status DIB with its associated status information, the supplemental status shall be held by the control unit until one of the following occurs:

a) it is read by the appropriate sense command; or

b) the associated status is reset.

The supplemental-status field shall be valid when the ES bit, bit 4 of the information-unit identifier, is set to one for a status DIB. Up to 32 bytes of sense information may be sent in the supplemental-status field. The contents and exact number of bytes in the supplemental-status field are model dependent (see 8.10.4.2).

When sense data is sent as supplemental status, it shall be sent in ascending order starting with sense-data byte 0 as the first byte of the supplemental-status field; it shall be sent in a single status DIB along with the status byte. The length of sense data sent in a status DIB with supplemental status shall be from a minimum of one byte to a maximum of 32 bytes. The number of bytes of sense data beyond one is model dependent.

Supplemental status shall not be used to report sense data associated with a resetting-event indication (see 9.6).

Supplemental status shall be sent in a single status DIB.

8.10.4.2 Sense Information

The data that is transferred during a basic sense operation or that is contained in the supplemental-status field of a status DIB when unit check is indicated, shall provide information concerning unusual conditions detected in a previous I/O operation and concerning the actual state of the device. Sense information shall provide more detailed information than the status byte and may describe reasons for the unit-check indication. It may also indicate that the device is in the not-ready state or that a drive is in the write-protected state.

Basic sense data shall not be reset as a result of a device executing the sense-ID command.

Bits 0-5 of the first sense-data byte (sense byte 0) shall be common to all devices. The six bits shall be independent of each other and, when set to ones, shall specify the events described in table 18. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 18 – Bits 0 - 5 of Sense-data Byte 0

Sense Bit	Description
0	Command Reject
1	Intervention required
2	Bus-out check
3	Equipment check
4	Data check
5	Overrun

Presentation of sense data as supplemental status is described in 9.2.3.2.

8.11 Control DIB Structure

8.11.1 Control DIB Structure Overview

The control DIB structure is used in all FC-SB-6 control IUs except for control IUs that specify the transport-confirm function (see 8.17). The control DIB structure and associated control functions shall not be used on a transport exchange.

A control DIB shall be sent only in a control IU by either a channel or control unit. It shall be used to transfer control information necessary for the execution of a CCW I/O operation or for device-level recovery. The structure of the control IU and its DIB is shown in figure 16. The control DIB consists of a 12-byte control header, a 4-byte LRC, and, for the purge-path-response function if error-code transfer is supported, a control payload and associated CRC field.

8.11.2 Control Header

8.11.2.1 Control Header Format

The control header shall be the first 12 bytes of a control DIB. The control header shall contain information indicating the control function to be performed and, for some functions, the parameters needed to perform the specified function. The control header shall consist of a one-byte control-function field, a three-byte control-parameters field, IU count field, a 2-byte control payload byte count field, and reserved bytes. The format of the control header shall be as shown in figure 36:

0	Control Function	Control Parameters	
1	Reserved		
2	Reserved	IU Count	Control Payload Byte Count
	0 7	15	31

Figure 36 – Control Header

Word 1 and byte 0 of word 2 shall be reserved and set to zeros by the sender and ignored by the recipient.

8.11.2.2 Control Function and Control Parameters

Byte 0 of word 0 shall contain the control function, which shall be interpreted in conjunction with the bits in the device-header-flags field of the IU header to determine the device-level function to be performed and the format of the control-parameter field. A summary of the functions represented by coding of bits 0-4 of the control-function field is shown in table 19. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 19 – Summary of Device-Level Control Functions

Bit Positions					
0	**1**	**2**	**3**	**4**	**Control Function**
0	0	0	0	0	Control end
0	0	0	1	0	Command response
0	0	1	0	0	Stack status
0	0	1	1	0	Cancel
0	1	0	0	0	System reset
0	1	0	1	0	Selective reset
0	1	1	1	0	Request status
1	0	0	0	0	Device-level exception
1	0	1	0	0	Status accepted
1	0	1	1	0	Device-level acknowledgment
1	1	0	0	1	Purge path
1	1	0	1	0	Purge path response
Note: All combinations of bits 0-4 that are not listed are reserved.					

If a reserved combination of bits 0-4 of the control-function field is used, a device-level protocol error shall be detected.

For all control DIBs, bits 5, 6, and 7 of the control-function field shall be set to zeros by the sender and are ignored by the recipient. table 20 provides a definition, in summary form, of the IUI bits and device-header flag bits for the various control functions. The recipient of a control DIB shall check the IUI bits and device-header flag bits for adherence to the description contained in the table. Bits that are required to be checked by the recipient which are set to a value other than specified by the table for the control function, shall result in a device-level protocol error being detected.

If the channel receives a control IU specifying a control function that only the channel is allowed to send and a control unit is allowed to receive (e.g., a cancel function), a device-level protocol error shall be detected. If a control unit receives a control IU specifying a control function that only a control unit is allowed to send and the channel is allowed to receive (e.g., a command response function), a device-level protocol error shall be detected.

Control IUs shall be sent as either FC-2 solicited or FC-2 unsolicited control information categories. A control IU that initiates a connection shall be sent as an unsolicited control information category. A control IU that does not initiate a connection shall be sent as a solicited control information category. Table 20 gives a summary of the control functions, and the sections following table 20 describe each control function and any control parameters required. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 20 – Summary of Control IUs

Control Function	IUI								Device-Header Flag Bits								FROM	F, M, or L	Control Parameter Field
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7			
				AS	ES	T3	T2	T1	E			CH	EE	CNP					
Control end	z	z	z	1	z	0	1	1	z	z	z	z	1	z	z	z	CH	M	z
Command response	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CU	F or M	z
Stack status	z	z	z	x	z	0	1	1	z	z	z	z	z	z	z	z	CH	FxL or L	z
Cancel	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CH	F or L	z
System reset	z	z	z	0	z	0	1	1	z	z	z	z	z	z	z	z	CH	F	z
Selective reset	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CH	F or L	*
Request status	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CH	F	z
Device-level exception: Address exception	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CU	FxL	$
Status accepted	z	z	z	x	x	0	1	1	z	z	z	x	z	z	z	z	CH	FxL or L	z
Device-level acknowledgment	z	z	z	x	z	0	1	1	z	z	z	z	z	z	z	z	CU	FxL or L	z
Purge path	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CH	F	$
Purge path response	z	z	z	1	z	0	1	1	z	z	z	z	z	z	z	z	CU	FxL	z

Explanation:
*Includes modifier bits in control-parameter field
$Includes exception or error code in control-parameter field
FStart new exchange pair. (First IU of exchange pair)
MMiddle IU of exchange. (Exchange pair remains)
LLast IU of a pre-existing exchange (Exchange pair removed)
FxLIU opens and closes an exchange.
CHChannel sends IU to control unit
CUControl unit sends IU to channel
xBit is set to one or zero as appropriate for conditions
zBit or field shall always be set to zero by the sender and is ignored by the recipient
0Bit shall always be set to zero and is checked for zero
1Bit shall always be set to one and is checked for one

8.11.2.3 Control End

The channel shall use the control-end function to indicate that it is unable to perform a command update for data chaining. A control IU indicating the control-end function is referred to as a control-end IU.

A control-end IU shall be sent only by a channel, and the AS bit and the EE bit shall always be set to one.

When the channel sends a control-end IU, it shall be indicating that, as a result of an abnormal condition, the transfer of the command update for data chaining was suppressed at the channel, and for a write operation, no more data is being sent for the current operation. A control-end IU sent by the channel shall require a status response from the control unit. The sending of additional IUs by the channel shall be held in abeyance until the required response is received.

A control unit that receives a control-end IU shall continue to execute the operation with the device until the conditions for ending an I/O operation are met and then it shall transfer status to the channel. The status DIB shall carry the residual count for the current CCW being executed when the control-end IU was received.

A control-end IU shall require both a valid CCW number and valid token. The CCW number used shall be the value the channel would have sent in the command update had it been able to perform the data chain. The token assigned is model dependent.

A control-end IU shall be sent as a solicited control FC-2 information category. It shall be sent as the middle sequence of an exchange.

8.11.2.4 Command Response

A control unit shall use a command-response function to indicate that execution of the first command of a CCW channel program has started as a non-immediate operation, or, a command IU containing both the CRR and CH bits set to one and the DU bit set to either zero or one, has become current. A control IU indicating command response is referred to as a command-response IU.

A command-response IU shall be sent only by a control unit. A command-response IU shall be sent only under the following conditions:

a) when the first command of the channel program is to be executed as a non-immediate command, or

b) when execution has started for a command IU, that is command-chained or data-chained, and for which the CRR bit in the command header is set to one.

The sending of a command-response IU shall not be contingent on sending or receiving any of the data associated with the command IU for which it is sent.

A command-response IU shall always require a valid CCW number and token, which shall be the CCW number and token received with the command for which the command response is being sent.

A command-response IU shall be sent as a solicited control FC-2 information category. It shall be sent as the first or middle sequence of an exchange, depending on whether there is an exchange already open for the same device on that logical path.

When the command-response IU is sent in response to the first command of a channel program, the least significant byte of the control-parameters field contains the 8-bit IU pacing parameter (see 8.10.3.4 for additional information).

8.11.2.5 Stack Status

The channel shall use the stack-status function to indicate that status is not accepted. The control unit or device shall hold the status information (status flags, status byte, and status parameter) and associated supplemental status, if any, until the status is requested by a request-status IU, until the status is presented as the response to a command IU, or until the status information is cleared or withdrawn. A control IU indicating stack status is referred to as a stack-status IU.

A stack-status IU shall be sent only by the channel. When a stack-status IU is received at a time when the inbound exchange has been closed with a data IU containing a status DIB, no response shall be sent; otherwise the response sent shall be a DACK IU.

The AS bit may be set to either one or zero in a stack-status IU. The setting of the AS bit in the stack-status IU shall be the same as the setting of the AS bit in the data IU containing a status DIB for which the stack-status IU is sent in response; otherwise, a device-level protocol error shall be detected.

A stack-status IU shall require a valid token but shall not require a valid CCW number. The CCW number field shall be meaningless for a stack-status IU. The token field for a stack-status IU shall not be checked by the recipient.

A stack-status IU shall be sent as a solicited control FC-2 information category. A stack-status IU shall be sent as the first and last sequence of an exchange if it is sent in response to an unsolicited data IU. It shall be sent as the last sequence if it is sent in response to a solicited data IU.

8.11.2.6 Cancel IU

The channel shall use the cancel function to cause the control unit to terminate the current CCW I/O operation, if any, for the specified device (see 9.4.3). A control IU indicating cancel is referred to as a cancel IU.

A cancel IU shall be sent only by the channel. When the cancel IU is received on the outbound exchange of an exchange pair at a time when the inbound exchange of the exchange pair has been closed, no response shall be sent; otherwise the response sent shall be a DACK IU.

A cancel IU shall require a valid token. A cancel IU shall not require a valid CCW number and the control unit shall ignore the CCW number. The acknowledgment sent in response to a cancel IU shall contain the CCW number and token associated with the current command for which the cancel function was performed and for which ending status is to be presented later.

A cancel IU sent during a connection shall be sent as a solicited control FC-2 information category, and shall be sent as the last sequence of the exchange, thus closing the outbound exchange. A cancel IU that initiates a connection shall be sent as an unsolicited control FC-2 information category, shall be sent as the first sequence of the exchange, and shall leave the exchange open. The outbound exchange shall be closed when the channel sends the LACK in response to the DACK.

NOTE 34 – A cancel function results in a disconnection when the DACK is received by the channel. In order to avoid having to save the active CCW numbers and tokens used during a connection and after a cancel function has been performed, the control unit indicates in the DACK sent, the CCW number and token for which the cancel was performed.

8.11.2.7 System Reset

The channel shall use the system-reset function to cause the control unit and associated devices to be reset with respect to the logical path on which the function was received (see 9.4.4). A control IU indicating system reset is referred to as a system-reset IU.

A system-reset IU shall be sent only by the channel and when recognized by the control unit the response sent shall be a DACK.

A system-reset IU shall require a valid token but shall not require a valid CCW number. The CCW number field for a system-reset IU shall be meaningless. The token field for a system-reset IU shall not be checked by the recipient. The token value received in the system-reset IU, shall be returned in the DACK response.

A system-reset IU shall always initiate a connection, shall be sent as an unsolicited control FC-2 information category, shall be sent as the first sequence of the exchange, and shall leave the exchange open. The outbound exchange shall be closed when the channel sends the LACK in response to the DACK.

8.11.2.8 Selective Reset

The channel shall use the selective-reset function to cause the control unit to end execution of the current CCW I/O operation, if any, for the specified device, and, depending on the bits within the control-parameter field, to perform one of the following functions (see 9.4.5 and 9.5.3):

a) selective reset;

b) channel-initiated retry; or

c) channel-initiated unit check.

A control IU indicating selective reset is referred to as a selective-reset IU.

A selective-reset IU shall be sent only by the channel. When the channel establishes initiative to perform a selective reset to a logical path and device and the channel has an open transport exchange with that device, the channel performs the transport-mode ABTS function (see 9.4.12) prior to sending a selective-reset IU to the device.

When the selective-reset IU is received on the outbound exchange of an exchange pair at a time when the inbound exchange of the exchange pair has been closed, no response shall be sent; otherwise the response sent shall be a DACK IU. When a selective reset IU which initiates a connection is received, the response sent shall be either a DACK IU or a data IU containing a status DIB (see 9.4.5 for additional information).

The control-parameter field for a selective-reset IU shall have the format shown in figure 37:

RC	0	0	RU	RO	Reserved	
0	1	2	3	4	5	23

Figure 37 – Control-parameter Field for the Selective-Reset IU

Bits 1, 2, and 5-23 shall be reserved.

When the RC bit, RU bit, and RO bit of the control-parameter field are all set to zeros, a selective reset shall be performed.

When one or more of the RC, RU, and RO bits of the control-parameter field are set to one and the requested function or functions are not able to be performed, a selective reset shall be performed.

When the RC bit of the control-parameter field is set to one, a channel-initiated retry shall be performed, if possible; if retry is not possible, the RU bit and RO bit of the control-parameter field shall determine the preferred method of terminating the I/O operation.

When either the RU bit or the RO bit of the control-parameter field is set to one and channel-initiated retry is either not requested or requested and not performed, a channel-initiated unit check shall be performed, if possible; if channel-initiated unit check is not performed, a selective reset shall be performed.

A selective-reset IU shall require a valid token and, depending on conditions at the channel and the setting of the RC, RU, and RO bits, a valid CCW number may or may not be required. If the RC, RU, or RO bit is set to one and the channel is requesting one of these functions to be performed for a specific CCW, then a valid CCW number shall be required. If the RC, RU, or RO bit is set to one but the channel is requesting that one of these functions be performed for the current CCW, a valid CCW number shall not be required and the value in the CCW number field shall be set to zero (see 9.5.3). If the RC, RU, and RO bits are all set to zero, a valid CCW number shall not be required. In this case the CCW number field shall be meaningless. If a selective-reset IU is sent after the control unit has disconnected and the token and CCW numbers have been reinitialized at the channel, a valid CCW number shall not be required and the CCW number field shall contain the value zero. The token field for a selective-reset IU shall not be checked by the recipient.

The DACK IU sent in response to a selective-reset IU shall contain the token and CCW number associated with the current command at the control unit which was terminated by the selective-reset function. If status is returned in a data IU as a result of performing either the request for retry or request for unit check functions, the token and CCW number associated with the command for which the function is being performed shall be used. If a disconnection occurs as a result of an abnormal condition and a selective reset is received as a result of recovery actions at the channel, the CCW number and token being maintained at the time of the disconnection shall be returned in the DACK. If, as a result of a normal disconnection from the channel, CCW numbers and tokens have been discarded, the DACK sent shall contain the token and CCW number received in the selective-reset IU.

Request Command Retry (RC): The RC bit, bit 0, when set to one, shall indicate that the channel is requesting the device to perform command retry on behalf of the channel.

If the RC bit is set to one, the device shall perform command retry if possible; if command retry is not able to be performed, the RU and RO bits shall determine the action to be taken. If the RC bit is set to zero, the RU and RO bits shall determine the action to be taken.

Request Unit Check (RU): Bit 3, when set to one, shall indicate that the channel is requesting the device to present unit check status. If the command is currently being executed by the device, the I/O operation shall be terminated and either unit check status shall be presented to the channel or the selective-reset operation performed.

Request Unit Check with Overrun (RO): Bit 4, when set to one, shall indicate that the channel is requesting the device to recognize an overrun condition and present unit check status. If the command is currently being executed by the device, the I/O operation shall be terminated and either unit check status shall be presented to the channel or the selective-reset operation performed.

The channel may set to one either the RO bit or the RU bit depending on the error condition. When command retry is either not requested or requested and not performed, the device shall interpret the RU and RO bits in the control-parameter field as shown in table 21:

Table 21 – Interpretation of the RO and RU Bits

RU	RO	Interpretation
0	0	A selective reset is performed
0	1	The device shall recognize an overrun condition and generate unit-check status with the sense data. Retry may be requested.
1	0	The device shall recognize a unit check and generate unit-check status with the appropriate sense data.
1	1	Error - the device shall ignore the requested functions and perform a selective reset.

A selective-reset IU that initiates a connection shall be sent as an unsolicited control FC-2 information category, shall be sent as the first sequence of the exchange, and shall leave the exchange open.

A selective reset shall be sent during a connection only if the RC, RU, and RO bits are all set to zero. It shall be sent as a solicited control FC-2 information category and as the last sequence of the exchange, closing the outbound exchange.

8.11.2.9 Request Status

The channel shall use the request-status function to indicate that the channel is prepared to have status information presented for the addressed device for which status is pending (see 9.4.2 and 9.4.6). A control IU indicating request status is referred to as a request-status IU.

A request-status IU shall be sent only by the channel.

The AS bit shall be set to one in a request-status IU.

For a request-status IU the CCW number and token assigned by the channel are model dependent.

A request-status IU shall always initiate a connection; it shall be sent as an unsolicited control FC-2 information category as the first sequence of the exchange.

8.11.2.10 Device-Level Exception

A control unit shall use the device-level-exception function to indicate that an abnormal condition was recognized in the IU received for which the sending of status is inappropriate or not permitted. The abnormal condition shall be indicated by the exception code present in the first byte of the control-parameter field. A control IU indicating device-level exception is referred to as a device-level-exception IU.

A device-level-exception IU shall be sent only by a control unit and may only be sent in response to an initiation IU.

A device-level exception shall have a control-parameter field with the format shown in figure 38.

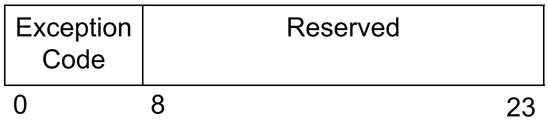


Figure 38 – Control-parameter Field for the Device-level Exception IU

Bits 8-23 of the control-parameter field shall be set to zeros by the sender of the device-level-exception IU and shall be ignored by the channel.

Bits 0-7 shall specify the abnormal condition detected in the IU for which this device-level-exception IU is the response. Table 22 shows code assignments for the exception codes. See 3.3.3 for a definition of the bit-numbering convention used in the table.

Table 22 – Exception Code Assignments

Bits								
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Exception Code**
0	0	0	0	0	0	0	0	Reserved
0	0	0	0	0	0	0	1	Address exception
0	0	0	0	0	0	1	0	Reserved
through								
1	1	1	1	1	1	1	1	

If a reserved exception code is used, a device-level protocol error shall be detected by the recipient of the device-level-exception IU.

A device-level-exception IU shall use the token and CCW number from the IU to which it is being sent in response.

Address Exception: Address exception shall indicate that the IU received contained the device address of an uninstalled device or a device that is offline with respect to the logical path. A device-level-exception IU for an address exception shall be sent only in response to certain IUs containing device-level information (see 9.5.4 for a description of the use of this exception code). A control IU indicating address exception is referred to as an address-exception IU.

An address-exception IU shall use the token, CCW number and device address from the IU to which it is being sent in response.

An address-exception IU shall be sent as a solicited control FC-2 information category, and it shall be sent as the first and last sequence of the exchange.

NOTE 35 – The device may be offline for some logical paths but not others. For example, the device may be precluded from going to the offline state if the offline switch was set to the offline position but the allegiances for some paths have not been cleared.

8.11.2.11 Status Accepted

The channel shall use the status-accepted function to indicate that the status received in a status DIB was accepted. A control IU indicating status accepted is referred to as a status-accepted IU.

A status-accepted IU shall be sent only by the channel. If the control unit did not close the inbound exchange with a data IU containing the status DIB, the response sent by the control unit shall be a DACK; otherwise no response is sent.

The CH bit in the device-header flags field for a status-accepted IU may be set to either one or zero, depending on conditions at the channel (see 9.4.8).

The AS bit may be set to either one or zero. The setting of the AS bit in the status-accepted IU shall be the same as the setting of the AS bit in the data IU containing a status DIB for which the status-accepted IU is sent in response; otherwise, a device-level protocol error shall be detected.

The ES bit may be set to either one or zero in a status-accepted IU. The ES bit shall be set to a one to indicate that the channel has accepted the supplemental status received in a status DIB with the ES bit set to a one. If the channel has not accepted the supplemental status received, or there was no supplemental status presented in the status DIB, the ES bit shall be set to zero (see 9.2.3.2).

The status-accepted IU shall require a valid token but does not require a valid CCW number. The token value used in the status-accepted IU is model dependent. The CCW number field and token field for a status-accepted IU shall not be checked.

A status-accepted IU shall be sent as a solicited control FC-2 information category. It shall be sent as the first and last, or last sequence of the outbound exchange, depending on whether there is an outbound exchange already open for that device or logical path.

8.11.2.12 Device-Level Acknowledgment (DACK)

The device-level acknowledgment (DACK) function shall indicate acceptance of a stack-status IU, cancel IU, system-reset IU, selective-reset IU, request-status IU, or status-accepted IU and that the requested function was or is to be performed. A control IU indicating a device-level ACK is referred to as a DACK IU.

A device-level acknowledgment shall be sent only by a control unit.

The AS bit may be set to either one or zero. The setting of the AS bit in the DACK IU shall be the same as the setting of the AS bit in the IU for which the DACK is sent in response; otherwise, a device-level protocol error shall be detected.

Except for a response to a cancel or selective-reset IU, a DACK IU shall use the token and CCW number from the IU to which it is being sent in response. When a DACK IU is sent in response to a cancel or selective-reset IU, the token and CCW number used shall depend on whether or not the cancel IU initiated a connection, conditions at the control unit and, for a selective-reset IU, the settings of the RC, RU and RO bits. See 8.11.2.8.

A DACK IU shall be sent as a solicited control FC-2 information category. It shall be sent as the first and last, or last sequence of the inbound exchange, depending on whether there is an inbound exchange already open for that device or logical path.

8.11.2.13 Purge Path

A channel shall use the purge-path function as part of device-level recovery in order to purge the transmit-receive path of any residual IUs associated with the logical path and device for which recovery is being performed; if error-code transfer is supported, the purge-path function shall also be used to transfer an error code to the control unit. A control IU indicating purge path is referred to as a purge-path IU. Support for error-code transfer is model-dependent.

A purge-path IU shall be sent only by a channel and when recognized by a control unit the response shall be a purge-path-response IU.

A purge-path IU shall require a valid token but shall not require a valid CCW number. The CCW number field for a purge-path IU shall be meaningless. The token value received in the purge-path IU shall be returned in the token field of the purge-path-response IU.

A purge-path IU shall be sent as the first sequence of an outbound exchange as an unsolicited control FC-2 information category.

The control-parameters field for the purge-path IU shall have the format shown in figure 39:

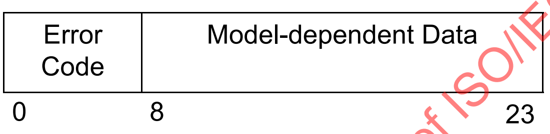


Figure 39 – Control-parameter Field for the Purge-Path IU

The error-code field contains a code designating the error which caused the channel to send the purge-path IU. The codes and their meanings shall be as shown in table 23.

Table 23 – Error Codes for the Purge-Path IU

Error Code (hex)	Meaning
00	Error-code transfer not supported
01	FC-SB-6 protocol timeout
02	FC-SB-6 link failure
03	Reserved
04	FC-SB-6 offline condition
05	FC-FS-4 link failure
06	FC-SB-6 length error
07	LRC error
08	FC-SB-6 CRC error
09	IU count error
0A	FC-SB-6 link-level protocol error
0B	FC-SB-6 device-level protocol error
0C	Receive ABTS
0D	Cancel function timeout
0E	Abnormal termination of exchange
0F	Host storage error
10	Software termination of exchange due to halt request
11	Software termination of exchange due to clear request
12	Interrogate operation error
13	Transport operation error
14	Transport error
15	REC error
16-FF	Reserved

When an error code of zero is indicated in the purge-path IU, indicating that error-code transfer is not supported, bits 8-23 of the control parameters field shall be reserved. When a nonzero error code is indicated in the purge-path IU, bits 8-23 of the control parameters field may contain model-dependent data to be transferred to the control unit.

8.11.2.14 Purge Path Response

A control unit shall use the purge-path-response function to indicate receipt and recognition of the purge-path function; if error-code transfer is supported, the purge-path-response function shall also be used to transfer an error code to the channel. A control IU indicating the purge path response is referred to as a purge-path-response IU.

A purge-path-response IU shall be sent only by a control unit in response to a purge-path IU. The channel shall send a LACK IU to close the outbound exchange when it receives a purge-path-response IU.

A purge-path-response IU shall be sent as a solicited control FC-2 information category, and shall be sent as the first and last sequence of the exchange.

When the purge-path-response IU is sent in response to a purge-path IU with a zero error code, indicating that the channel does not support error-code transfer, then no control payload shall be included.

When the purge-path-response IU is sent in response to a purge-path IU containing a nonzero error code, then a control payload shall be included only if the control unit supports error-code transfer. The format of the control payload is shown in figure 40:

Neighbor-F_Port LESB	Control-Unit-N_Port LESB	Error Code	Model-dependent Data	
0	24	48	49	255 (max)

Figure 40 – Control Payload Format for the Purge-Path-Response IU

If the control unit is attached to an F_Port and has successfully obtained the LESB of the attached F_Port, the neighbor-F_Port LESB field, bytes 0-23 of the control payload, shall contain the link error status block (LESB) of the attached F_Port; if the control unit is directly attached to the N_Port of a channel or is unable to obtain the LESB of the attached F_Port, then the neighbor F_Port-LESB field shall be set to zero.

The control-unit-N_Port LESB field, bytes 24-47 of the control payload, contains the LESB of the N_Port of the control unit.

The error code field, byte 48 of the control payload, contains an 8-bit code indicating the error which the control unit has recognized for the logical path and device indicated in the purge-path IU. The codes and their meanings shall be as shown in table 24.

.

Table 24 – Error Codes for the Purge-Path-Response IU

Error Code (hex)	Meaning
00	No errors were recognized
01	FC-SB-6 protocol timeout
02	FC-SB-6 link failure
03	Logical-path-timeout error
04	FC-SB-6 offline condition
05	FC-FS-4 link failure
06	FC-SB-6 length error
07	LRC error
08	FC-SB-6 CRC error
09	IU count error
0A	FC-SB-6 link-level protocol error
0B	FC-SB-6 device-level protocol error
0C	Receive ABTS
0D	Reserved
0E	Abnormal termination of exchange
0F	Logical path not established
10	Test initialization result error
11-FF	Reserved

Bytes 49-255 of the purge-path-response IU may be used to transfer up to 207 bytes of model-dependent data to the channel. If fewer than 207 bytes of model-dependent data are transferred, the control payload is truncated to include bytes 0-49 plus the actual number bytes of model-dependent data transferred.

8.11.3 Control Payload

Support for the control payload field is model-dependent. When a control payload is supported, the control payload shall be a variable length field with a minimum length of 49 bytes, and a maximum length of 256 bytes. The control payload, if supported, shall be present only in a purge-path-response IU sent in response to a purge-path IU with a nonzero error code. See 8.11.2.14.

8.12 Link-Control DIB Structure

8.12.1 Link-Control DIB Structure Overview

A link-control DIB shall be sent in a control IU by the channel or control unit. It shall be used to transfer FC-SB-6 link-level information. A control IU containing a link-control DIB is referred to as a link-control IU.

A link-control IU shall be used to perform the following link-level functions (see 6.4):

a) establish a logical path;

b) remove a logical path;

c) test initialization; or

d) acknowledge, reject or indicate a busy condition.

A link-control DIB shall contain a link header, an LRC, and for some functions a link payload with its associated CRC field (see Figure 16).

8.12.2 Link Header

8.12.2.1 Link Header Format

The link header shall be generated by the channel or control unit and shall appear immediately after the IU header of either a solicited or unsolicited control IU. The link header shall contain information indicating the link-control function to be performed and, for some functions, the parameters needed to perform the specified function. The link header shall consist of a one-byte link-control-function field, a two-byte link-control-information field, a two-byte CTC counter field, a one-byte IU count, a two-byte link payload byte count, and reserved bytes. The link header shall have the format shown in figure 41.

0	Reserved	Link Control	Link Control Information	
1	CTC Counter		Reserved	
2	Reserved	IU Count	Link Payload Byte Count	
	0		16	31

Figure 41 – Link Header

Byte 0 of word 0, bytes 2 and 3 of word 1, and byte 0 of word 2 shall be reserved and set to zero by the sender and ignored by the recipient. For all IUs except the ELP IU, bytes 0 and 1 of word 1 are also set to zero by the sender and ignored by the recipient.

8.12.2.2 Link Control

The link-control field, byte 1 of word 0, shall contain the link-header format bits, bits 5-7, and the link-control function bits, bits 0-4. The link control field shall have the format shown in figure 42.

Link-control Function					Link Header Format		
0	1	2	3	4	5	6	7

Figure 42 – Link-Control Field

When bits 5-7 are set to the value ‘001’b, bits 0-4 of the link-control field shall contain the link-control function. See 6.4 for the link-control functions and their assigned codes. All other values of bits 5-7 shall be reserved. If a value of other than ‘001’b is specified, a link-level protocol error shall be recognized.

A link-control function, when sent as a request by the channel, shall require a valid token and when sent as a request by a control unit, the token value shall be meaningless. A link control response sent by a control unit shall use the token and CCW number from the IU to which it is being sent in response. A link control response sent by a channel shall have a valid token assigned by the channel (but not a valid CCW number) and shall not use the token and CCW number from the IU to which it is being sent in response.

8.12.2.3 Link-Control Information

The link-control information field, bytes 2 and 3 of word 0, shall contain additional parameters necessary in order to perform the link-control function specified by bits 0-4 of byte 1 of word 0. The parameters shall be determined by the link-control function specified.

The link-control information field shall be meaningful only when the ELP, LPE, LPR, extended TIN, TIR, and LRJ link-control functions are specified (see 6.4 for a description of the parameters).

The parameters associated with the LPR and LRJ link-control functions shall be contained in the first byte of the link-control information field, byte 2 of word 0. The second byte of the link-control information field, byte 3 of word 0, shall always be set to zero by the sender and ignored by the recipient.

The parameters associated with the ELP and LPE link-control functions shall be contained in both bytes of the link-control information field, bytes 2 and 3 of word 0.

The parameters associated with the TIN and TIR link-control IUs shall be contained in bit 7 of byte 0 and in byte 1 of the link-control information field. When bit 7 of byte 0 is zero for the TIN IU, the TIN function is specified; when bit 7 of byte 0 is one, the TINC function is specified. When bit 7 of byte 0 is zero for the TIR IU, the TIR function is specified; when bit 7 of byte 0 is one, the TINCR function is specified. Bits 0-6 of byte 0 of the link-control information field are reserved.

8.12.2.4 Channel-to-Channel (CTC) Counter

The CTC counter field, bytes 0 and 1 of word 1, is only meaningful when ELP is the link-control function specified, and bit 15 of the link-control information field is set to one. The CTC counter specifies the number of CTC connections for which the channel sending the ELP IU is providing emulated control unit functionality at the time when the ELP IU is sent.

8.12.3 Link Payload

The link payload shall be a variable length field with a maximum length of 8156 bytes. The link payload field shall be present only for the TIR link-control function (see 6.4.8).

8.13 Transport Command IU

8.13.1 Transport Command Overview

A transport-command IU shall be sent as an unsolicited command IU and shall be sent only by the channel. The IU is used to initiate an operation with a device and to transfer information associated with the current TCW to the control unit.

The transport-command IU is made up of an 8-byte FC-SB-6 header followed by a 4-byte transport-command header (TCH), a 16-byte transport-command-area header (TCAH), a variable length transport-command area (TCA), a 4-byte LRC, a 4-byte data-transfer length (DL), and for bidirectional operations, a 4-byte bidirectional read data-transfer length (BRDL).

The structure of the transport-command IU is shown in figure 43.

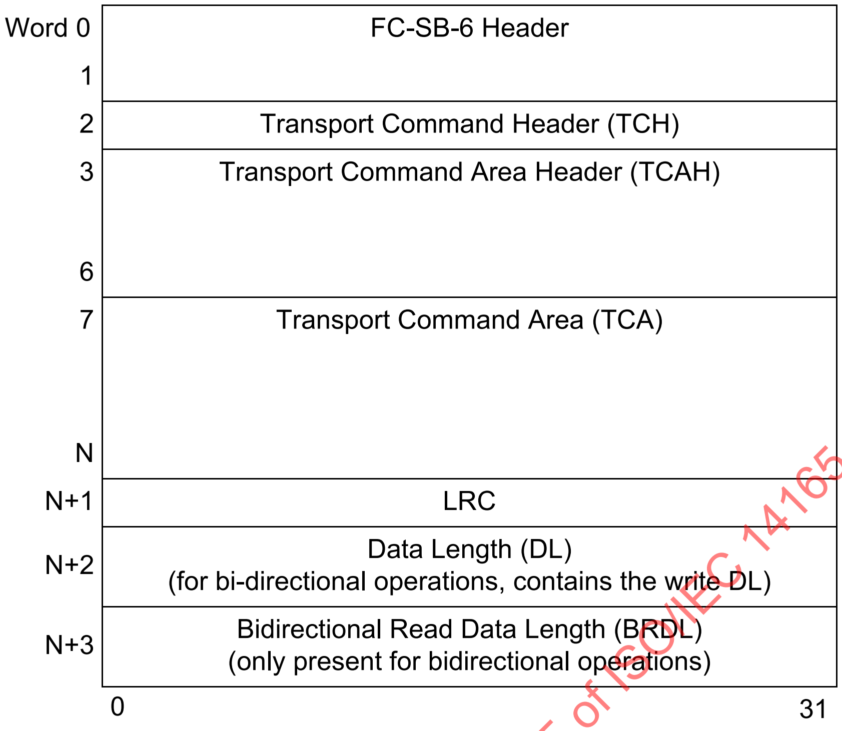


Figure 43 – Transport-Command IU

The combination of the TCAH, TCA, LRC, DL and, for bidirectional operations, the BRDL together are referred to as the transport-command-control block (TCCB).

8.13.2 FC-SB-6 Header

The FC-SB-6 header is as defined in 8.4. The FC-SB-6 header provides FC-4 addressing information to identify the logical path and the device for the transport-command IU.

8.13.3 Transport Command Header

8.13.3.1 Transport Command Header Format

The transport command header (TCH) contains 4 bytes and immediately follows the FC-SB-6 header in a transport-command IU. The TCH contains information about the TCCB and the associated device operations. The format of the TCH is shown in Figure 44.

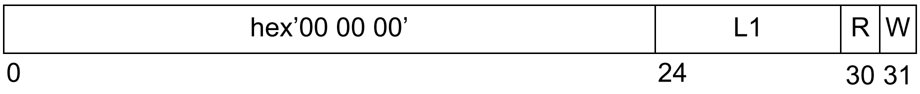


Figure 44 – Transport Command Header (TCH)

Bytes 0-2 shall be set to hex ‘00 00 00’ by the channel and shall be ignored by the control unit.

8.13.3.2 L1

The L1 field, bits 24-29, specifies the length, in words, of the TCA plus the 1-word LRC field that directly follows the TCA.

For control units that do not support bidirectional operations, the total amount of data transferred in the transport-command IU shall be equal to the L1 field plus 8 (that is, the L1 field plus the 2-word FC-SB-6 header, the 1-word TCH, the 4-word TCAH and the 1-word DL); otherwise, the control unit shall recognize a transport-command IU integrity error due to a data count error.

For control units that support bidirectional operations, the total amount of data transferred in the transport-command IU shall be equal to the L1 field plus 8 (that is, the L1 field plus the 2-word FC-SB-6 header, the 1-word TCH, the 4-word TCAH and the 1-word DL) or the L1 field plus 9 (that is, the L1 field plus the 2-word FC-SB-6 header, the 1-word TCH, the 4-word TCAH, the 1-word DL, and the 1-word BRDL for a bidirectional operation); otherwise, the control unit shall recognize a transport-command IU integrity error due to a data count error.

If a transport-command IU integrity error is not recognized, the location of the LRC in the TCCB is located based on the L1 field and the LRC check can be performed (see 8.13.6).

NOTE 36 – see the definition of the L2 field (8.13.4.3) for further information on the L1 field and the relationship to the L2 field in the TCAH.

8.13.3.3 Read (R)

The R bit, bit 30, when set to one, indicates that the TCA contains one or more commands that transfer read data. When the bit is set to zero, the TCA does not contain any commands that transfer read data. When the R bit is one and the W bit (bit 31) is zero, the operation is referred to as a read operation.

When both the R and W bits are set to zero, the TCA shall not contain any commands that transfer read or write data indicating that the I/O operation does not perform data transfer. When both the R and W bits are set to one, the TCA may contain commands that transfer both read and write data, and the operation is referred to as a bidirectional operation. The device shall recognize a TCH content error due to a read-write conflict if a bidirectional operation is specified and the control unit does not support bidirectional operations, or the control unit supports bidirectional operations but the transport-command IU does not contain the BRDL field.

When the R bit is zero and the device encounters a command in the TCA that attempts to transfer read data, the device shall recognize a TCCB content error due to an invalid read attempt.

8.13.3.4 Write (W)

The W bit, bit 31, when set to one, indicates the TCA includes one or more commands that transfer write data. When the bit is set to zero, the TCA does not include any commands that transfer write data. When the W bit is one and the R bit (bit 30) is zero, the operation is referred to as a write operation.

For a description of when both the W and R bits are set to zero, and when both the W and R bits are set to one, see 8.13.3.3.

When the W bit is zero and the device encounters a command in the TCA that specifies a write command, the device shall recognize a TCCB content error due to an invalid write condition.

8.13.4 Transport Command Area Header

8.13.4.1 TCA Header Format

The TCA header (TCAH) contains 16 bytes and immediately follows the TCH in a transport-command IU. The TCA header contains information about the TCA and the associated device operations. The format of the TCA header is shown in Figure 45.

0	Format Control	hex ‘00’	Reserved	
1	Reserved			L2
2	Service Action Code		Reserved	I/O Priority
3	Reserved			
	0	8 13	16	24 31

Figure 45 – TCA Header Format

Byte 1 of word 0 shall be zero; otherwise, a TCCB content error shall be recognized by the control unit. Bytes 2-3 of word 0, bytes 0-2 of word 1, byte 2 of word 2 and word 3 are reserved and ignored by the control unit.

8.13.4.2 Format Control

Byte 0 of word 0 contains the format control field and shall be equal to hex’7F’; otherwise, a TCCB content error shall be recognized by the control unit.

8.13.4.3 L2

Byte 3 of word 1 is the L2 field and contains an unsigned binary integer that shall specify the length, in bytes, of the TCA plus words 2 and 3 of the TCA header, plus the 4-byte LRC field that directly follows the TCA. The L2 field shall be a nonzero multiple of 4 that is between 20 and 252, inclusive.

Bits 6 and 7 of the L2 field shall be zero; otherwise a TCCB content error shall be recognized by the control unit. Additionally, the L2 field shall specify a length that is 8 bytes greater than the length specified by the L1 field in the TCH (see 8.13.3.2); otherwise, a TCCB content error shall be recognized by the control unit.

NOTE 37 – The definition of L1 length field is based on the FCP-3 and SPC-4 definition of a command IU and command-descriptor block (CDB) which defines the L1 location in the command IU to contain the “additional CDB length”. The CDB itself begins at word 3 of the command IU (i.e., beginning with the format-control field) and extends through word 6. According to FCP-3, the “additional CDB” portion begins at word 7 of the command IU (i.e., beginning with the TCA) and extends through the LRC, and L1 is defined to contain the length of this additional CDB portion.

NOTE 38 – The L2 length field is based on the SPC-4 definition of a CDB and defines the L2 location in the command IU to specify a different “additional CDB length”. This additional CDB length is defined to begin at word 5 of the command IU (i.e., beginning with the service action code field) and extends through the TCA and LRC. Since the length specified by L2 begins at word 5 of the command IU and the length specified by L1 begins at word 7 of the command IU, L2 always exceeds L1 by exactly 8 bytes.

8.13.4.4 Service Action Code

Bytes 0-1 of word 2 contain an unsigned integer value that specifies the type of operation specified by the TCCB as follows:

Hex Value	Meaning
1FFE	FC-SB-6 Device I/O operation.
1FFF	FC-SB-6 Device support functions.

8.13.4.5 I/O Priority Number

Byte 3 of word 2 contains the control-unit I/O priority number. The number is an unsigned binary integer that specifies the priority level that is applied at the control unit during execution of a TCW channel program. I/O priority provides a means for specifying an end-to-end priority. The usage of I/O priority is beyond the scope of this standard.

The specified control-unit I/O priority number may be any value in the range of 1 to 255. The number 1 shall designate the lowest priority that may be assigned to the I/O-operations at the control unit and the number 255 shall designate the highest priority that may be assigned. The number 0 shall mean that no priority is assigned to the associated I/O operations. The handling of I/O-operations, in this case, shall depend upon the control unit model.

Depending upon the model, fewer than 255 priority levels may be supported by the control unit in which case, the control unit shall round the specified priority to the equivalent level of priority that it supports. For example, if the control unit supported 127 levels of priority, then a specification of 254 would result in an equivalent priority of 127. Also depending upon the model, the control unit may not support the function of I/O priority, in which case, byte 3 of word 2 shall be ignored.

NOTE 39 – A control unit should implement a model-dependent fairness algorithm which ensures that no I/O operation is permanently prevented from completing because it is setting the I/O priority to a lower value relative to the I/O priority used by other I/O operations.

8.13.5 Transport Command Area

8.13.5.1 Transport Command Area Format

The transport-command area (TCA) is a variable length area that contains from 1 to 30 device-command words (DCW) and any associated DCW control data. The maximum size of the TCA is 240 bytes. The TCA has the format shown in Figure 46.

0 1	DCW	
2	DCW or Control Data for Previous DCW	
	DCW or Control Data for Previous DCW	
	. . .	
N	DCW or Control Data for Previous DCW	
	0	31

Figure 46 – Transport-Command Area (TCA) (Where N>0 and N<=59)

The length of the TCA is an integral number of 4-byte words. When sufficient space exists in the TCA for the entire DCW, a DCW begins on the word boundary that follows the previous DCW or the control data associated with the previous DCW. When the last DCW in the TCA specifies control data that is not an integral number of words, the subsequent LRC word begins on the word boundary that follows the control data.

8.13.5.2 DCW Format

The DCW is an 8-byte control block that is designated on an word boundary. The format of the DCW is described in Figure 47.

0	Command	Control Flags	Reserved	CD Count
1	DCW Data Count			
	0	8	16	24 31

Figure 47 – Device Command Word

Byte 2 of word 0 is reserved and shall be set to zero by the channel and ignored by the control unit.

8.13.5.2.1 Command

Byte 0 of word 0 is the command field for the DCW and shall contain an 8-bit code that specifies a device operation to be performed. The basic operations shall be specified by the following commands: read, write, control, sense, and transport.

Some commands contain modifier codes that specify model-dependent command execution.

Table 25 lists the basic commands and the bit settings of the command field.

Table 25 – Contents of the DCW Command Field

Command	Bit Position							
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Reserved	M	0	M	M	0	0	0	0
Transport[1]	0	1	M	M	0	0	0	0
Reserved	1	1	M	M	0	0	0	0
Sense	M	M	M	M	0	1	0	0
Reserved	M	M	M	M	1	0	0	0
Reserved[2]	M	M	M	M	1	1	0	0
Write	M	M	M	M	M	M	0	1
Read	M	M	M	M	M	M	1	0
Control	M	M	M	M	M	M	1	1

Explanation:

M represents a modifier bit and may be set to zero or one.

Notes:

[1] These codes are reserved for commands used to manage transport-mode operations.

[2] In command mode, these codes are used for the read backward commands - read backward commands are not supported in transport mode.

The command field may contain any value that is not reserved. If a reserved command code is sent in a DCW, the device shall recognize a TCCB content error.

Some commands, when executed, do not result in the transfer of data but cause the device to chain and start execution of the next command when all of the conditions for command chaining are satisfied. Each of the basic operations is described in the following section.

Read: A read command shall initiate execution of a device operation that performs device-to-channel data transfer. The bytes of data within a block shall be provided in the same order as those received by the write command.

Write: A write command shall initiate execution of a device operation that performs channel-to-device data transfer.

Control: A control command shall initiate execution of a device operation that makes use of control data provided in the DCW.

Sense: The sense command is similar to a read command, except that the data shall be obtained from sense indicators rather than from a record source.

Transport: A transport command specifies a device support function associated with transport-mode operations. It may contain control data and may perform a read or write data transfer.

There are some commands (particular combinations of eight bits in the command field) which shall be executed by all devices. Table 26 indicates these required commands.

Table 26 – Required Commands

Command Code (hex)	Function
04	Basic Sense
03	No-operation (no-op)
md	Read configuration data
md	Read node identifier
E4	Sense ID
md	Set interface identifier
Explanation: md Model-dependent command code. The command code may be obtained by using a sense-ID command.	

The defined transport commands are shown in table 27.

Table 27 – Transport Commands

Command Code (hex)	Function
40[1]	Interrogate
50[2]	Transfer TCA Extension (TTE)
60[3]	Transfer CRC Offset Block (TCOB)
70[3]	Transfer extended CRC Offset Block (TeCOB)

Notes:

[1] Support for the interrogate command is required when transport mode is supported.

[2] Support for the TTE command is required to support more DCWs than can be accommodated in the TCA.

[3] Support for the TCOB and/or TeCOB command is not required but recommended when intermediate CRC is used in transport mode write data.

In command mode, transport command codes are defined as reserved and result in unit check status (see 8.7.2.2). The transport commands are defined in the following clauses.

8.13.5.2.1.1 Interrogate Command

The interrogate command is used to determine the state of a TCW I/O operation at a device. The command does not initiate an operation at the device, does not affect the state of the device and does not reset allegiances associated with the device and logical path. See 9.3.3.2 for details of the interrogate operation.

The incorrect-length condition is not recognized for the interrogate command regardless of whether the DCW-incorrect-length facility is supported by the channel and control unit.

8.13.5.2.1.2 Transfer CRC Offset Block and Transfer Extended CRC Offset Block Commands

The Transfer CRC Offset Block (TCOB) or Transfer Extended CRC Offset Block (TeCOB) command transfers a CRC Offset Block or an Extended CRC Offset Block respectively, to a control unit and, when used, shall be the first DCW of the TCA. The location of the COB or eCOB is determined by the control-data count and may be provided in either the first write transport-data IU of an I/O operation or as control data following the TCOB or TeCOB DCW of an I/O operation. If the control-data count is non-zero then the COB or eCOB shall be provided as control data following the TCOB or TeCOB DCW. If the control-data count is zero, then the COB or eCOB shall be provided in the first write transport-data IU of the operation. When a COB or eCOB is provided in a transport-data IU, the DCW data count shall include a byte count of the COB or eCOB intermediate-CRC-offset fields but shall not include the COB or eCOB CRC or COB or eCOB pad bytes.

Use of the TCOB or TeCOB command is recommended in TCW I/O Operations whenever intermediate CRC is present in write data. A device indicates support for the TCOB and/or TeCOB command using model dependent configuration commands. A TCOB or TeCOB command shall not be sent to a device that does not indicate support for this command. The chain-command flag in a DCW containing the TCOB or TeCOB command shall be set to one.

The CRC Offset Block (COB) specifies the location of intermediate CRC words within the transport data for a write data transfer. A COB is provided when the first DCW in the TCA contains the TCOB command. If the first DCW in the TCA does not contain the TCOB command, then a COB is not provided.

The CRC Offset Block (COB) has the format shown in Figure 48.

0	Intermediate CRC 0 offset
1	Intermediate CRC 1 offset
.	.
.	.
N	Intermediate CRC N offset
	0 or 1 word of pad (if the COB is in a transport-data IU)
	CRC word (if the COB is in a transport-data IU)

Figure 48 – CRC Offset Block

The Extended CRC Offset Block (eCOB) specifies the location of intermediate CRC words within the transport data for a write data transfer and the number of pad bytes preceding the CRC word. Bits 0-29 of the eCOB entry shall contain the word offset of an intermediate CRC word within the transport data for a write data transfer, and bits 30 to 31 shall contain the number of pad bytes. An eCOB is provided if the first DCW in the TCA contains the TeCOB command. If the first DCW in the TCA does not contain the TeCOB command, then an eCOB is not provided.

The Extended CRC Offset Block (eCOB) has the format shown in Figure 49.

0	Intermediate CRC 0 offset	#pad bytes
1	Intermediate CRC 1 offset	#pad bytes
.	.	
.	.	
N	Intermediate CRC N offset	#pad bytes
	0 or 1 word of pad (if the eCOB is in a transport-data IU)	
	CRC word (if the eCOB is in a transport-data IU)	
	0 29	30 31

Figure 49 – Extended CRC Offset Block

The COB or eCOB shall contain 1 to N+1 entries of intermediate-CRC offsets or intermediate-CRC offsets and pad lengths, respectively. If the COB or eCOB is provided in a transport-data IU, then the COB or eCOB shall also contain 0 or 1 word of pad and 1 word of CRC. Zero or one word of pad shall be provided so that the word containing the CRC will be on a word boundary that is not on a doubleword boundary. The last word of the COB or eCOB shall contain a CRC that covers the entries 0 through N. The CRC shall be generated as described in 8.6.5. If the COB or eCOB is provided as control data for the TCOB or TeCOB DCW, then there shall be no COB or eCOB padding or CRC and the control-data count shall be equal to 4 times the number of intermediate-CRC-offset fields in the COB or eCOB.

Each intermediate-CRC offset in the COB or eCOB indicates the relative offset from the first byte of data in the transport-data IU. If the COB or eCOB is provided in a transport-data IU, the first byte of data for the write transfer is either the first byte following the COB or eCOB CRC if a TCAX is not being provided or the first byte following the TCAX CRC if a TCAX is being provided (see 8.13.5.2.1.3). If the COB or eCOB is provided in the TCA as DCW control data, the first byte of data for the write transfer is either the first byte in the transport-data IU if a TCAX is not being provided or the first byte following the TCAX CRC if a TCAX is being provided.

If the control unit determines that one or more entries in the COB or eCOB indicate that an intermediate CRC is not at a location that is appropriate for the device and/or command being executed, then it may present program check status. If program check status for this condition is presented it shall indicate I/O Exception Reason Code 9.

The incorrect-length condition is not recognized for the transfer-CRC-offset-block or transfer-extended-CRC-offset-block command regardless of whether the DCW-incorrect-length facility is supported by the channel and control unit.

NOTE 40 – The presence of a TCOB DCW or TeCOB DCW or TTE DCW does not affect DCW chaining requirements of a device. For the purpose of enforcing device DCW chaining rules, the DCW following the TCOB or TeCOB or, if present, the DCW following the TTE, shall be considered the first in the chain.

8.13.5.2.1.3 Transfer TCA Extension (TTE)

The Transfer TCA Extension (TTE) command transfers a TCA extension to a control unit. The TCAX is considered a logical extension of the TCA. When neither a TCOB DCW nor a TeCOB DCW is present in the TCA, the TTE DCW shall be the first DCW in the TCA. When a TCOB DCW or TeCOB DCW is present in the TCA, the TTE DCW shall be the second DCW in the TCA. The TTE DCW data count specifies the length of the TCAX and shall be an integral multiple of four bytes. The TTE DCW control-data count shall be zero and the chain-command flag shall be set to one. The DCW data count shall not include the TCAX CRC or TCAX pad bytes.

Use of the TTE command is required to be able to transfer more DCWs and control data than can be accommodated in the TCA provided in the transport-command IU. The TCAX contains additional DCWs or both additional DCWs and control data for the TCW I/O operation. A device indicates support for the TTE command using model dependent configuration commands. A TTE command shall not be sent to a device that does not indicate support for the TTE command and shall only be sent when there is at least one DCW in the TCA that does not contain a transport command (see 8.13.5.2.1).

Control data for a DCW in the TCA shall not extend beyond the last byte in the TCA - control data is not permitted to span the TCA and TCAX.

The chain command flag in the last DCW in the TCA shall be set to one when a TCAX has been specified.

The TCAX has the format shown in Figure 50.

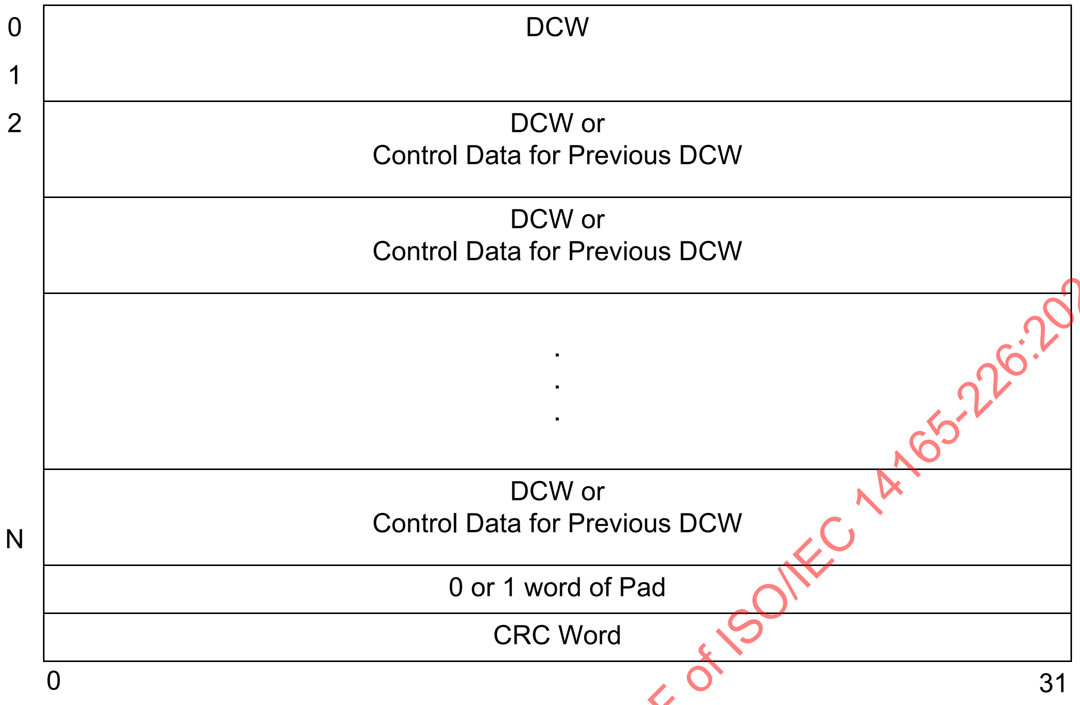


Figure 50 – Transport-Command Area Extended (TCAX) plus Pad Bytes and CRC

DCWs and control data in the TCAX are processed as defined for the TCA (see 9.3.3, “TCA Processing”). The maximum length of the TCA plus the TCAX shall be less than 64K bytes.

The TCAX shall be followed by either zero pad bytes or one word of pad, followed by one word of CRC. One word of pad is provided if the size of the TCAX is an even multiple of 4 bytes; otherwise no pad bytes are provided. The total size of the TCAX plus the word of pad, if provided, plus the one word CRC shall be an even multiple of 4 bytes The TCAX CRC covers the entire TCAX and the pad word, if provided. The CRC shall be generated as described in 8.6.5.

The incorrect-length condition is not recognized for the transfer-TCA-extension command regardless of whether the DCW-incorrect-length facility is supported by the channel and control unit.

NOTE 41 – The limitation of the length of the TCA plus TCAX to less than 64K bytes is due to the 2-byte DCW offset field provided in the transport-response IU (see 8.15.6.4).

8.13.5.2.2 Control Flags

Byte 1 of word 0 is the control-flags field for the DCW. The format of the field is as shown in figure 51. Bits 0 and 3-7 are reserved and shall be set to zero by the channel and ignored by the control unit.

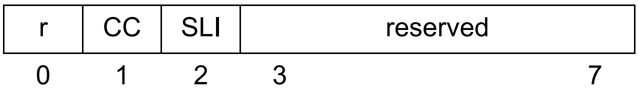


Figure 51 – DCW Control Flags

Chain Command (CC): The CC bit, bit 1, when set to one, shall specify an intent to perform chaining of commands. Upon normal completion of the current device operation and after recognition of device end at the device, chaining of commands shall cause the next DCW in the TCA or TCAX to be executed. The CC bit shall be set to zero in the last DCW in the TCA.

Suppress Length Indication (SLI): When the DCW-incorrect-length facility is supported by the channel and control unit, bit 2 is the SLI flag and shall control whether command chaining is to occur on an incorrect-length condition and whether incorrect length is to be indicated in the transport-response IU on an incorrect-length condition. An incorrect-length condition is detected by the control unit if the DCW data count does not match the amount of data required by the device for a write DCW or if the DCW data count does not match the amount of data available at the device for a read DCW. When the DCW-incorrect-length facility is not supported by the channel and control unit, bit 2 is reserved and shall be set to zero by the channel and ignored by the control unit.

When the SLI flag is one and an incorrect-length condition exists for the current DCW, command chaining, if indicated, shall be permitted. If the SLI flag is zero and an incorrect-length condition exists for a DCW, command chaining, if indicated, shall not be permitted and, if no abnormal conditions are detected, ending status with the DCW offset and DCW residual count for the current DCW and with the incorrect-length flag (see 8.15.4.11) set to one shall be transferred to the channel.

When an incorrect-length condition exists for a DCW and the SLI flag in the DCW is equal to one and no unusual conditions exist for the current DCW, data transfer shall be performed as follows; otherwise, the operation shall be terminated and an abnormal termination condition shall be reported for the operation (see 9.3.4). When an incorrect-length condition exists for a DCW and the SLI flag in the DCW is equal to zero, data transfer shall be performed as specified in 8.15.4.11 in the description of the incorrect-length flag.

Data transfer for a read DCW shall be performed as follows when an incorrect-length condition exists and the SLI flag is one in the DCW:

a) If the data count is greater than the amount of data available at the device for the command, then:

   1) If the CC flag is one, the data available at the device shall be transferred to the channel and pad bytes shall be transferred so that the amount of data transferred for the DCW is equal to the data count. Pad bytes shall be set to zero. All data transferred, including pad bytes, shall be included in the computation of CRC required for the operation; or

   2) If the CC flag is zero, the data available at the device shall be transferred to the channel and either no additional data shall be transferred or pad bytes shall be transferred so that the amount of data transferred for the DCW is equal to the data count. Pad bytes shall be set to zero. The data transferred plus any pad bytes shall be included in the computation of CRC required for the operation. It is model dependent whether pad bytes are transferred for this case.

b) If the data count is less than the amount of data available at the device for the command, only an amount of data equal to the data count shall be transferred to the channel for the DCW. Only the data transferred shall be included in the computation of CRC required for the operation.

Data transfer for a write DCW shall be performed as follows when an incorrect-length condition exists and the SLI flag is one in the DCW:

a) If the data count is greater than the amount of data required by the device for the command, then:

   1) If the CC flag is one, the amount of data specified by the data count shall be transferred to the control unit. The data transferred shall be used in the computation of CRC required for the operation; data not required by the device is discarded; or

   2) If the CC flag is zero, the amount of data required by the device shall be transferred to the control unit. Additional data shall be transferred up to the next intermediate CRC word or until the data count for the DCW is exhausted. All data transferred shall be used in the computation of CRC required for the operation.

b) If the data count is less than the amount of data required by the device for the command, the amount of data specified by the data count shall be transferred to the control unit. The data transferred shall be used in the computation of CRC required for the operation.

8.13.5.2.3 Control-Data (CD) Count

Byte 3 of the DCW is the CD count and contains an 8-bit unsigned binary integer that shall specify the number of control data bytes that immediately follow this DCW in the TCA.

If the command-code specifies a command that requires control data and byte 3 of word 0 contains zeros, a TCCB content error shall be recognized.

If the CD count contains a value that specifies control data past the end of the TCA or the value specifies data past the end of the TCAX, a TCCB content error shall be recognized.

8.13.5.2.4 Data Count

Word 1 is the DCW data-count field and contains a 32-bit unsigned binary integer that shall indicate the number of bytes to be transferred between the channel and control unit during execution of the DCW, not including any pad and CRC bytes.

8.13.6 Longitudinal Redundancy Check

The longitudinal-redundancy-check (LRC) field shall a contain a 32-bit LRC code which immediately follows the TCA in the transport-command IU (see Figure 43). LRC shall be generated by the originator and recipient using a seed of hex’A5 5A A5 5A’ that is exclusive or’ed on a word by word basis over the following fields:

a) The FC-SB-6 header;

b) The TCH, TCAH and TCA;

c) The DL field; and

d) The BRDL field (when present).

If the value of the LRC contained in the IU equals the value generated by the recipient, a valid LRC shall be recognized; otherwise, an invalid LRC shall be recognized. If a valid LRC is recognized, the contents of the transport IU shall be considered valid; otherwise, the IU shall be considered invalid and a transport-command-IU integrity error due to invalid LRC shall be recognized.

See Annex C for an example of the process used for LRC calculation.

8.13.7 Data Length

For a read operation, the data-length (DL) field contains a 4-byte unsigned binary integer that specifies the number of bytes to be transferred by the control unit to the channel for the operation and includes any pad bytes required to round to a word boundary if not already on a word boundary plus 4 bytes for CRC.

For a write operation or bidirectional operation, the DL field shall contain a 4-byte unsigned binary integer that specifies the number of bytes to be transferred by the channel to the control unit for the operation and includes all intermediate and final pad and CRC bytes, and if a COB or eCOB is included in the first transport-data IU of the write operation, it also includes the COB or eCOB, any COB pad bytes or eCOB pad bytes and the COB CRC or eCOB CRC bytes. If a TTE DCW is present in the TCA, the DL field also includes the TCAX, any TCAX pad bytes and TCAX CRC bytes.

8.13.8 Bidirectional Read Data Length

For a bidirectional operation, the bidirectional-read-data-length (BRDL) field shall contain a 4-byte unsigned binary integer that specifies the number of bytes to be transferred by the control unit to the channel for the operation and includes any pad bytes required to round to a word boundary if not already on a word boundary plus 4 bytes for CRC.

For operations that are not bidirectional operations, the device shall recognize a TCH content error due to a read-write conflict if the transport-command IU contains the BRDL field.

8.14 Transport Data IU

8.14.1 Transport Data IU Overview

A transport-data IU shall be sent only in a solicited data IU by either the channel or control unit. When the transport-data IU is sent by the channel, sequence initiative for the exchange shall be transferred to the recipient. When the transport-data IU is sent by the control unit, sequence initiative for the exchange shall be held by the control unit.

For a write or bidirectional operation, the channel shall send one or more transport-data IUs to the control unit to transfer the write data for the operation. When first-transfer-ready is disabled, the channel shall send a transport-data IU immediately following the transport-command IU. For all transport-data IUs from the channel after the first transport-data IU and for all transport-data IUs from the channel when first-transfer-ready is not disabled, the channel shall send a write transport-data-IU only after receiving a transfer-ready IU. The channel shall send a transport-data IU after each transfer-ready IU is received until all write data specified by the TCCB is transferred or a transport-response IU is received. (See 8.14.2 and 9.3.2.2 for additional information on write data transfer).

For a read or bidirectional operation, the control unit shall send at least one transport-data IU to the channel to transfer the read data specified for the operation. (See 8.14.2 and 9.3.2.3 for additional information on read data transfer).

For bidirectional operations, when first-transfer-ready disabled is not in effect, the control unit selects the first transport-data IU to be transferred. The control unit either sends a transfer-ready IU to the channel to request a transport-data IU or sends a transport-data IU to the channel. If first-transfer-ready disabled is in effect for the operation, the channel send a transport-data IU following the transport-command IU.

8.14.2 Transport Data

For a write or bidirectional operation, when first-transfer-ready disabled in not in effect, the maximum number of bytes allowed in a transport-data IU sent by the channel is specified by the most recent transfer-ready IU. When first-transfer-ready is disabled, the maximum number of bytes allowed in the first transport-data IU sent by a channel prior to receiving a transfer-ready IU is as specified in 9.3.2.2.

For all subsequent transport-data IUs, the maximum number of bytes allowed is specified by the most recent transfer-ready IU. All transport-data IUs sent by the channel shall contain the maximum number of bytes allowed except for the last transport-data IU.

For a read or bidirectional operation, the maximum number of bytes allowed in a transport-data IU is 4 gigabytes minus 4. For read operations, the control unit shall not transfer an amount of data that exceeds the amount specified by the DL field in the transport-command IU. For bidirectional operations, the control unit shall not transfer an amount of data that exceeds the amount specified by the BRDL field in the transport-command IU.

When the transport-data IU containing the last data byte for the current TCCB contains an amount of data between the maximum-allowable amount of data for the IU minus 3 and the maximum-allowable amount of data for the IU, the CRC field shall not be included in the transport-data IU and shall be sent in a subsequent transport-data IU by itself. The transport-data IU containing the last data with an amount of data up to the maximum amount of data minus 3 bytes up to the maximum amount of data shall have pad bytes added, if necessary, and the CRC value sent in the subsequent transport-data IU shall cover both the data and the pad bytes.

8.14.3 Pad Bytes

Pad bytes, if present, are contained in the last word of a data area for which CRC is to be calculated and shall be used to pad the data area to the next word boundary when the data area to be covered is not an integral number of data words. This applies to data areas to be covered by COB or eCOB, intermediate and final CRC.

The value used for a pad byte is model dependent.

8.14.4 Cyclic-Redundancy-Check

8.14.4.1 Final CRC

The final CRC field is included in a transport-data IU when the transport-data IU is the last transport-data IU sent by the channel or control unit for the TCW I/O operation. The final-cyclic-redundancy-check (CRC) field shall contain a word-aligned 32-bit redundancy-check code.

8.14.4.2 Intermediate CRC

One or more intermediate CRC words may be present in a transport-data IU sent by the channel. The intermediate CRC words provide CRC checking of data areas prior to transmission of the entire data area specified by the TCA. The method for inserting and detecting intermediate CRC fields is outside the scope of this document when neither the TCOB nor TeCOB command is provided.

8.14.4.3 COB CRC

COB CRC is present in a transport-data IU sent by the channel when a COB is being transferred in the transport- data IU (see Figure 48). The COB CRC follows the COB intermediate-CRC-offset fields and COB padding, if present.

8.14.4.4 eCOB CRC

eCOB CRC is present in a transport-data IU sent by the channel when an eCOB is being transferred in the transport- data IU (see Figure 49). The eCOB CRC follows the eCOB intermediate-CRC-offset fields and eCOB padding, if present.

8.14.5 CRC Generation and Checking

For a write or bidirectional operation, when intermediate CRC has not been sent by the channel, the channel shall generate final CRC on all data sent by the channel for the operation. If COB, eCOB, or intermediate CRC has been provided in a transport-data IU, the final CRC shall be generated only over the data transferred after the last intermediate CRC field. If a read or bidirectional operation completes without:

a) device status containing unit check; or

b) a non-zero transport response exception code,

then the control unit shall generate CRC for all data sent by the control unit.

The recipient shall generate CRC on the data received and compare the value calculated to the CRC contained in the IU. If the value of the CRC contained in the IU equals the value calculated by the recipient, a valid CRC check shall be recognized; otherwise, an FC-SB-6 CRC error shall be recognized.

The processing of CRC, in terms of generation and checking, shall follow the equations in Annex A of FC-FS-4; however, the coefficients of the polynomial representing the transport-data field are not chosen according to bit transmission order specified in Annex A. The coefficient of the highest order term of F(x), which is the polynomial representing the transport-data field, shall represent the most-significant bit of the transport-data field; lower-order coefficients of the polynomial F(x) correspond to less-significant bits of the transport-data field in sequential order. The 32-bit CRC shall be the 32 coefficients of the frame check sequence (FCS) polynomial in Annex A, equation (1). The most significant bit of the CRC shall be the coefficient of the highest order term of the FCS polynomial; sequentially less-significant bits shall be the coefficients of sequentially lower order terms of the FCS polynomial. The CRC is transmitted on the link in the same bit-transmission order as all other words of the transport-data field. Additional information may be found in Annex B of Reference[8].

The initialized value of the CRC generator shall be hex’FFFFFFFF’.

8.15 Transport Response IU

8.15.1 Transport Response IU Overview

A transport-response IU shall be sent as a command status FC-FS-4 information category and shall be sent only by the control unit. The IU provides status for a TCW I/O operation. The type of status may include normal ending status or, when an abnormal condition has been detected, termination status that indicates the cause for abnormal termination of the operation. The type of status being presented is specified within the transport-response IU. The IU may include extended status that provides further status for the operation.

A transport-response IU may or may not close the transport exchange. Whether or not the exchange has been closed by a transport-response IU is indicated in the FC-FS-4 header (see FC-FS-4). If the transport exchange has not been closed by the transport-response IU, the channel shall send a transport-confirm IU (see 8.17) that closes the exchange after receiving the transport-response IU.

The control unit shall request status confirmation from the channel that a transport-response IU has been received by sending the transport-response IU such that it does not close the exchange and transfers sequence initiative to the channel.

The control unit shall request status confirmation when the transport-response IU contains a resetting-event notification (see 9.6) or when the device status contains control-unit-busy status (see 9.4.10); otherwise, a device-level protocol shall be recognized.

NOTE 42 – For performance reasons, the preferred implementation for a transport-response IU that contains CE and DE-only or CE-only device status is that the IU close the transport exchange.

8.15.2 Transport Response IU Structure

A transport-response IU shall consist of an FC-SB-6 header followed by a 20-byte status field, a 4-byte status LRC, and an optional extended-status field containing from 32 to 64 bytes. When extended status is provided, a 4-byte extended-status LRC field is provided as the last word of the transport-response IU. Pad bytes are added to the extended status to round to the next word boundary if the number of extended-status bytes is not on a word boundary. The structure of the transport-response IU is shown in Figure 52.

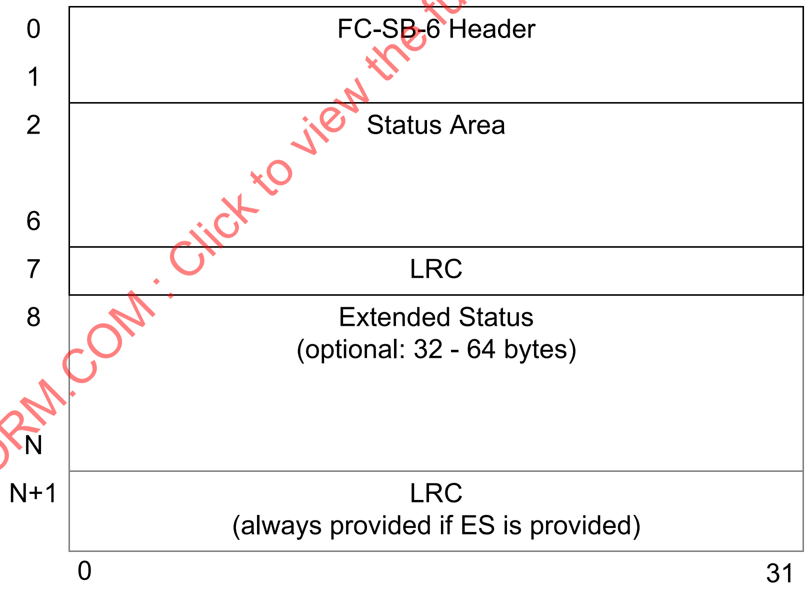


Figure 52 – Transport Response IU

8.15.3 FC-SB-6 Header

The FC-SB-6 header has the format defined in 8.4 and is set equal to the FC-SB-6 header received in the transport-command IU for this exchange.

The FC-SB-6 header in a transport-response IU is not used to identify the transport exchange at the channel - the channel uses the FQXID (see 4.7.4) to identify the exchange.

8.15.4 Status

8.15.4.1 Status Area Format

The status area is 20 bytes and immediately follows the FC-SB-6 header in a transport-response IU. The status contains information about the TCW I/O operation. The format of the status is shown in Figure 53.

0	Status Flags1	Max CU Exchanges Parameter	Response Flags	Response Code
1	DL Residual Count (for bidirectional operations, contains the write DL residual count)			
2	MFTBC		Response Length	
3	Bidirectional Read DL Residual Count (for non-bidirectional operations, this field is reserved)			
4	hex ‘7F’	Status Flags2	Status Flags3	Device Status
	0	8	16	24 31

Figure 53 – Transport Response IU Status Area Format

When word 3 does not contain a bidirectional read DL-residual count, the word shall be set to zeros by the control unit and ignored by the channel.

8.15.4.2 Status Flags1

Byte 0 of word 0 contains the status-flags1 field. The control unit shall use this field to provide information about the associated TCW I/O operation. Bits 1-3 shall be set to zeros by the control unit and shall be ignored by the channel. The format of the status flags1 shall be as shown in Figure 54.

IL	reserved	Exception Code
0	1 3	4 7

Figure 54 – Transport Response Status Flags1

Incorrect Length (IL): When the DCW-incorrect-length facility is supported by the channel and control unit, bit 0 is the IL flag and, when set to one, shall indicate that the TCW I/O operation was terminated due to an incorrect-length condition for the DCW indicated by the DCW offset (see 8.15.6.4). The IL flag is set to one only when the SLI flag is zero and when the device status includes channel-end status.

When the DCW-incorrect-length facility is not supported by the channel and control unit, bit 0 is reserved and shall be set to zero by the control unit and shall be ignored by the channel.

When the IL flag is set to one in a transport-response IU, the DCW offset identifies the DCW containing the incorrect-length condition and the DCW residual count indicates the amount of data transfer, if any, that was not transferred for the incorrect-length DCW. Data transfer, if any, for DCWs that preceded the incorrect-length DCW in the TCA and, if present, the TCAX shall have completed and all CRC checking on transferred data shall have been performed as follows:

a) If the DCW containing the incorrect-length condition is a read DCW:

    1) The last transport-data IU sent to the channel shall contain CRC for all read data transferred to the channel during the TCW I/O operation.

    2) If write DCWs preceded the incorrect-length read DCW in the TCA and, if present, the TCAX, the data for those write DCWs shall have been received by the control unit and CRC checking shall have been performed on the data. The control unit shall request write data for DCWs that follow the incorrect-length DCW as necessary to obtain the CRC required to perform the CRC checking.

b) If the DCW containing the incorrect-length condition is a write DCW:

    1) Any data for the incorrect-length write DCW (as indicated by the residual count) and all data for preceding write DCWs in the TCA and, if present, the TCAX shall have been transferred to the control unit and CRC checking shall have been performed on the data. The control unit shall request write data for DCWs that follow the incorrect-length DCW as necessary to obtain the CRC required to perform the CRC checking.

    2) If read DCWs preceded the incorrect-length write DCW in the TCA and, if present, the TCAX, and CRC for the read data has not yet been sent to the channel, a transport-data IU shall be sent to the channel that contains CRC for all read data transferred to the channel during the TCW I/O operation.

Exception Code: bits 4-7 contains an exception code. The code shall be set by the control unit to report an abnormal condition detected during a TCW I/O operation. The control unit shall set this code to zero when it provides interrogate extended status in a transport-response IU. The codes are defined in table 28.

Table 28 – Transport Response Exception Codes

Code (hex)	Meaning
0	No exception condition detected by the control unit.
1	Device-level exception due to an address-exception condition (see 9.5.4.2).
2	Link-level reject due to a logical-path-not-established condition (see 6.4.9).
3	Resetting event notification - a resetting event (see 9.6) has occurred on the logical path and device associated with the transport-command IU. When this code is set in the transport-response IU, the control unit shall request status confirmation for the status. If status confirmation is received, the resetting event condition is reset at the device for the logical path; otherwise, the resetting-event condition remains pending.
4	Device-detected program check/IFCC - the control unit has detected a condition that may result in a program check or IFCC to be reported. Errors that fall into this category include errors that indicate the transport-command IU arrived in a corrupted state, invalid CRC detected for write data, and receipt of a second I/O operation for a logical path and device address that is not an interrogate operation.
5	Device-detected program check - the control unit has detected an error in the content of the TCH (see 10.9.3) or TCCB (see 10.9.4).
6-F	Reserved

8.15.4.3 Maximum CU Exchanges Parameter

Byte 1 of word 0 shall contain the maximum-CU-exchanges parameter. The parameter, when nonzero, contains an unsigned binary integer that, when added to one and multiplied by 16, shall specify the maximum number of transport exchanges the channel can have open at the control unit. When the parameter is zero, the maximum number of exchanges is not updated and remains at the current value for the maximum number of transport exchanges supported by the channel for the control unit. The default value for the maximum number of transport exchanges supported by the channel for the control unit is 64.

8.15.4.4 Response Flags

Byte 2 of word 0 shall contain the response flags and shall be set to hex’02’.

8.15.4.5 Response Status Code

Byte 3 of word 0 shall contain the response-status code and shall be set to hex’00’ by the control unit.

8.15.4.6 Data Length Residual Count

When the DLRCV bit in the status-flags2 field is one, word 1 shall contain the DL-residual count as follows:

a) For write and bidirectional operations, the DL-residual count is a 32-bit, unsigned binary integer that specifies the difference between the DL field in the transport-command IU and the number of bytes actually received from the channel.

b) For read operations, the DL-residual count is a 32-bit, unsigned binary integer that specifies the difference between the DL field in the transport-command IU and the number of bytes actually sent to the channel.

When the DLRCV bit in the status-flags2 field is zero, this field is meaningless.

8.15.4.7 Maximum First Transfer Buffer Credit (MFTBC)

The MFTBC field, bits 0-15 of word 2, is valid when the maximum-first-transfer-buffer-credit valid (MFTBV) bit in Status Flags2 is one and, when valid, shall contain a 16-bit unsigned binary integer that indicates the maximum number of first-transfer buffers supported by the control unit for use by the channel.

When the MFTBV bit is zero, the MFTBC shall be set to zero by the control unit and shall be ignored by the channel.

8.15.4.8 Response Length

Bits 16-31 of Word 2 shall contain the response length (RL). The RL is a 16-bit unsigned binary integer that shall specify the number of bytes provided in the transport-response IU starting with word 6 until the end of the transport-response IU. The value includes any extended status and extended-status LRC that is provided in the transport-response IU. The RL shall have a minimum of hex’08’ and a maximum of hex’4C’.

8.15.4.9 Bidirectional Read DL Residual Count

For bidirectional operations when the DLRCV bit in the status-flags2 field is one, word 3 shall contain the BRDL-residual count. For bidirectional operations, the BRDL-residual count is a 32-bit, unsigned binary integer that specifies the difference between the BRDL field in the transport-command IU and the number of bytes actually sent to the channel.

For operations that are not bidirectional operations or for bidirectional operations when the DLRCV bit in the status-flags2 field is zero, this field is reserved.

8.15.4.10 Sense Type Code

Byte 0 of word 4 shall contain the sense-type code and shall be set to hex’7F’.

8.15.4.11 Status Flags2

Byte 1 of word 4 shall contain the status-flags2 field. The control unit shall use the field to provide additional information about the associated TCW I/O operation. Reserved bits shall be set to zeros by the control unit and shall be ignored by the channel. The format of the Status-flags2 field is shown in Figure 55.

	DLRCV	IS	LE	MFTBV	Reserved
Bit	0	1	2	3	4 7

Figure 55 – Status Flags2

DL Residual Count Valid (DLRCV): the DLRCV bit, bit 0, when set to one indicates that the DL-residual count and, for bidirectional operations, the BRDL-residual count in the status area are valid. Bit 0, when set to zero, indicates the DL-residual count and, for bidirectional operations, the BRDL-residual count are not valid. Bit 0 shall be set to one when device status is presented containing channel-end without unit-check.

Initial Status (IS): the IS bit, bit 1, when set to one, shall indicate that the transfer-response IU contains status for a condition that was recognized prior to attempting to perform the first DCW in the TCA or that the operation was executed as an immediate operation and DCW chaining was not indicated for the TCA.

Logable Event (LE): the LE bit, bit 2, shall be set to one to request that the channel store vendor specific log information associated with the operation when the control unit detects an unusual condition that may require manual intervention. The method by which the log information is stored is vendor specific.

Maximum First Transfer Buffer Credit Valid (MFTBV): When the use of first-transfer-buffer credits (see 9.3.2.2.2) is in effect, the MFTBV bit, bit 3, shall be set to one to indicate that the maximum-first-transfer-buffer credit (MFTBC) field in the status area contains a valid MFTBC. When bit 3 is set to zero, the MFTBC field does not contain a valid MFTBC. If the use of first-transfer-buffer credits is not in effect, bit 3 shall be set to zero by the control unit and shall be ignored by the channel.

8.15.4.12 Status Flags3

Byte 2 of word 4 shall contain the Status-flags3 field. The control unit shall use this field to provide additional information about the associated transport-mode operation. The format of the Status-flags3 field is shown in figure 56.

<table><tr><td></td><td colspan="2">Reserved</td><td colspan="2">TCRR Code</td><td>ESS</td></tr><tr><td>Bit</td><td>0</td><td>4</td><td>5</td><td>6</td><td>7</td></tr></table>

Figure 56 – Status Flags3

Bits 0-4 shall be set to zeros by the control unit and shall be ignored by the channel.

Transport Mode Command Retry Reason (TCRR): When transport-command retry is requested by the control unit (see 9.5.2.2), bits 5-6 shall contain a 2-bit code that provides information regarding the reason for the transport-command retry. When transport-command retry is not being requested by the control unit, bits 5-6 shall be set to zeros by the control unit and shall be ignored by the channel.

The following TCRR codes are defined:

Code	Meaning
0	Reserved.
1	Retry requested for unspecified reason.

2 Retry requested due to a first-transfer buffer shortage at the control unit for a transport-data IU received from the channel.

3 Reserved.

Extended Status Sent (ESS): The ESS bit, bit 7, shall be set to one to indicate that extended status, including possible sense data, has been sent in the transport-response IU. When this bit is one and the response-length field in word 2 is less than hex’2C’ or greater than hex’4C’, a device-level protocol error is recognized. When this bit is zero and the response-length field is greater than 8, a device-level protocol error is recognized.

8.15.4.13 Device Status

Byte 3 of word 4 shall contain the device-status. The device status byte indicates device and control unit status and shall have the format shown in table 15.

The device status shall be set to zero by the control unit in a transport-response that contains interrogate status and shall be ignored by the channel.

If the channel receives a transport-response IU for the completion of an I/O operation that has no exception condition indicated and the device status is equal to zero, the channel shall recognize a device-level protocol error. A device-level protocol error may also be recognized if the combination of status is not appropriate for the existing conditions. When an exception condition is indicated, device status shall be set to zero by the control unit and shall be ignored by the channel.

8.15.5 Status LRC

The status LRC is 4 bytes and immediately follows the status in a transport-response IU (see Figure 52). The LRC shall be generated by the originator and recipient using a seed of hex’A5 5A A5 5A’ that is exclusive or’ed on a word by word basis over the following fields in the transport-response IU:

a) FC-SB-6 header

b) Status (words 2-6)

If the value of the status LRC contained in the IU equals the value generated by the recipient, a valid LRC shall be recognized; otherwise, an invalid LRC shall be recognized. If a valid LRC is recognized, the content of the transport-response IU shall be considered valid; otherwise, the content of the transport-response IU shall be considered invalid and a transport-response status LRC error shall be recognized.

See Annex C for a description of the LRC procedure and an example of an LRC calculation.

8.15.6 Extended Status

8.15.6.1 Extended Status Format

When the ESS bit is set in the Status Flags3 field, extended status has been provided in the transport-response IU. The minimum size of the extended status, when provided, shall be 32 bytes.

Extended status immediately follows the status LRC in a transport-response IU. The field contains additional status and information about the TCW I/O operation, the associated device operation and possibly sense data or error information. The general format of the extended-status field is shown in Figure 57 - the ES-type code in the ES flags specifies the format of the type-dependent area.

<table>
<tr><td>0</td><td>ES Length</td><td>ES Flags</td><td colspan="2">DCW Offset</td></tr>
<tr><td>1</td><td colspan="4">DCW Residual Count</td></tr>
<tr><td>2</td><td colspan="4">Reserved</td></tr>
<tr><td>3</td><td colspan="4" rowspan="2">Type-Dependent Area
(N less than or equal to 15)</td></tr>
<tr><td>N</td></tr>
<tr><td></td><td>0</td><td>8</td><td>16</td><td>31</td></tr>
</table>

Figure 57 – Extended Status General Format

8.15.6.2 Extended Status Length

Byte 0 of word 0 shall contain the extended-status length (ESL). The ESL is an 8-bit unsigned binary integer that shall indicate, in bytes, the amount of data in the extended status. The field shall contain a minimum value of 32 and a maximum value of 64. The extended-status LRC is not included in the ES length.

8.15.6.3 Extended Status Flags

Byte 1 of word 0 shall contain the extended-status (ES) flags. The flags contain validity indicators, information about the I/O operation and specifies the type of extended status being provided. The format of the ES flags is shown in Figure 58.

DOV	RCV	CM	TV	r	ES Type Code	
0	1	2	3	4	5	7

Figure 58 – Extended Status Flags

Bit 4 shall be set to zero by the control unit and ignored by the channel.

DCW Offset Valid (DOV): The DOV bit, bit 0, when set to one, shall indicate that the DCW-offset field contains a valid DCW-offset value. When bit 0 is zero, the DCW-offset field is not meaningful.

DCW Residual Count Valid (RCV): When the DOV bit is one, the RCV bit, bit 1, when set to one, shall indicate that the DCW-residual-count field contains a valid DCW-residual count for the DCW specified by the DCW-offset field. When the DOV bit is zero, the RCV bit is meaningless. This bit shall be set to one when channel end status is provided without unit check status and no exception condition indicated.

Cache Miss (CM): The CM bit, bit 2, when set to one, shall indicate that one or more I/O-device cache misses occurred during the I/O operation. When the CM bit is zero, no cache misses occurred during the I/O operation. The CM bit provides information relative to the performance of the operation.

Timers Valid (TV): The TV bit, bit 3, is valid when the ES-type code specifies I/O status, and when valid, shall indicate, when set to one, that the timer fields provided in words 3-7 of the extended status area are valid. When bit 3 is zero, or the ES-type code does not specify I/O status, then valid timer information has not been provided. When the device status contains control-unit busy status, the TV bit shall be set to zero by the control unit.

ES Type Code: the ES-type code, bits 5-7, shall specify the type of information that has been provided in the extended status area. The codes are as defined in table 29.

Table 29 – Extended Status Type code

Code	Meaning
0	Reserved.
1	I/O status - the extended-status area contains valid ending status for the transport-mode I/O operation.
2	I/O Exception - the extended-status area contains information regarding termination of the transport-mode I/O operation due to an exception condition.
3	Interrogate status - the extended-status area contains status for an interrogate operation.
4-7	Reserved.

8.15.6.4 DCW Offset

When the DOV bit in the extended-status flags is one, the value in bytes 2-3 of word 0, if less than the size of the TCA, shall be the byte offset, from the beginning of the TCA, of the last DCW in the TCA that was attempted at the control unit. When the DOV bit is one and the value in bytes 2-3 of word 0 is greater than or equal to the size of the TCA, the value in bytes 2-3 of word 0 minus the size of the TCA is the offset, from the beginning of the TCAX of the last DCW in the TCAX that was attempted at the control unit. When the DOV bit in the extended-status flags is zero, bytes 2-3 of word 0 are not meaningful.

When all DCWs in the TCA and, if present, in the TCAX are executed without abnormal conditions, the DOV bit shall be set to one and the DCW offset shall be set for the last DCW in the TCA. When TCA processing cannot be completed due to an abnormal condition, the DCW offset field indicates the point at which DCW processing was terminated.

When the DOV bit is one, the residual-count-valid (RCV) bit in the extended status is valid and, if set to one, indicates that the DCW-residual count contains a valid residual count for the DCW specified by the DCW offset.

8.15.6.5 DCW Residual Count

When the DOV and RCV flags in the ES-flags are both one, word 1 shall contain a 32-bit unsigned binary number that specifies the difference between the DCW data count for the DCW specified by the DCW offset field and the actual number of bytes either written to a device or read from a device and sent to the channel for the DCW. Pad and CRC bytes are excluded from the residual count calculation. When either the DOV or RCV flags is zero, word 1 is not meaningful.

For a write DCW, if the device requires less data than specified by the count for a DCW, the residual count shall be equal to the difference between the DCW data count and the actual number of bytes of data used. If the device requires more data than specified by the DCW data count, the residual count shall be zero for this DCW.

For a read DCW, if less data than is specified by the count for a DCW is transferred to the channel, the residual count shall be equal to the difference between the DCW data count and the actual number of bytes sent. If all of the data specified by the count for DCW is sent but more data would have been sent to the channel had a larger count been specified, the residual count shall be equal to zero.

Note that the residual count for a DCW is greater than zero only when the record length is less than the DCW data count.

The following situations may require additional flags to be set in the status or may prevent including a valid residual count in a transport-response IU.

a) If a unit check condition prevents determining the exact amount of data either read from or written to a device, a valid residual count shall not be calculated. In this case, the RCV bit shall be set to zero in the transport-response IU. Unit-check status shall be indicated along with channel-end status;

b) If an abnormal condition as indicated by the exception code in the transport-response IU (see table 28) prevents determining the exact amount either read from or written to a device, then a valid residual count shall not be calculated. In this case, the RCV bit shall be set to zero and a non-zero exception code shall be set in the transport-response IU; or

c) If during a read operation, a unit check condition prevents determining the exact amount of data transferred to the channel for a DCW, then a valid residual count shall not be calculated. In this case, the RCV bit shall be set to zero and unit check along with channel-end status shall be indicated.

8.15.6.6 I/O Status Extended Status Format

When the ES-type code in the ES flags specifies the I/O-status type code, the format of the extended status shall be as shown in Figure 59.

0	ES Length	ES Flags	DCW Offset
1	DCW Residual Count		
2	Reserved		
3	Device Time		
4	Defer Time		
5	Queue Time		
6	Device Busy Time		
7	Device-Active-Only Time		
8 N	Additional Data (when ES Length > 32)		
	0	8	16 31

Figure 59 – I/O Status Extended Status Format

Words 0-2 are as defined in 8.15.6.1.

8.15.6.6.1 Device Time

When the TV bit in the ES flags is one, word 3 shall contain a 32-bit binary unsigned integer that indicates the device time for the TCW I/O operation. The device time begins to accumulate when either the transport-command IU has been accepted for execution at the control unit (see 9.3.1) or the control unit begins to accumulate device-busy time, and continues to accumulate until the control unit sends a transport-response IU containing ending status for the operation. The resolution of bit 31 of the device time is one microsecond. The frequency offset to nominal frequency of the oscillator used to calculate the time interval shall not exceed 150 x 10 $^{-6}$. The sum of the defer time, queue time, device-busy time and device-active-only time shall not exceed the device time.

When the device time is zero, either no accumulation shall have occurred, a failure of the timing facility shall have been recognized, or an overflow condition shall have been recognized.

If a timing facility failure has occurred, the control unit shall send a transport-response IU with unit check status and sense information describing the failure to allow device-dependent program recovery. If a device-time overflow condition has been recognized, unit check status shall not be presented to the channel. When a timing facility failure has been recognized and after unit-check status has been presented, the control unit shall continue with execution of subsequent TCW channel programs as before; however, it shall present a device time of zero. In this case, unit-check status shall not be presented after the initial indication of a timing facility failure.

When the TV bit in the ES flags is zero, word 2 does not contain device time information. When unit-check status is provided as the result of a timing-facility failure, the TV bit shall be set to zero in the ES flags.

8.15.6.6.2 Defer Time

When the TV bit in the ES flags is one and the defer-time function is provided by the control unit, word 4 contains a 32-bit binary unsigned integer that indicates the defer time for the TCW I/O operation. The resolution of bit 31 of the defer time is one microsecond. The frequency offset to nominal frequency of the oscillator used to calculate the time interval shall not exceed 150 x 10 $^{-6}$.

Defer time shall be accumulated by the control unit whenever the TCW I/O operation needs to be temporarily delayed because a device-dependent action needs to be performed. This condition typically occurs whenever the control unit requires the device to access the physical medium for the purpose of reading or writing data. However, because of the device-dependent nature of the action, other conditions may also be recognized by the control unit creating the need for a temporary delay of the TCW I/O operation. In all cases, device-defer time shall continue to be accumulated with each successive temporary delay until execution of the TCW I/O operation has been completed.

When the defer time is zero, either no accumulation shall have occurred, the defer-time function is not supported, a failure of the defer-timing facility shall have been recognized, or an overflow condition shall have been recognized.

If a defer-timing facility failure has occurred, the control unit shall send a transport-response IU with unit check status and sense information describing the failure to allow device-dependent program recovery. If a defer-timing overflow condition has been recognized, unit check status shall not be presented to the channel. When a defer-timing facility failure has been recognized and after unit-check status has been presented, the control unit shall continue with execution of subsequent channel programs as before; however, it shall present a defer time of zero. In this case, unit-check status shall not be presented after the initial indication of a defer-timing facility failure.

When the TV bit in the ES flags is zero, word 3 does not contain defer time information. When unit-check status is provided as the result of a timing-facility failure, the TV bit shall be set to zero in the ES flags.

8.15.6.6.3 Queue Time

When the TV bit in the ES flags is one and the queue-time function is provided by the control unit, word 5 contains a 32-bit binary unsigned integer that indicates the queue time for the TCW I/O operation. The resolution of bit 31 of the queue time is one microsecond. The frequency offset to nominal frequency of the oscillator used to calculate the time interval shall not exceed 150 x 10^{-6}.

Queue time represents the time the I/O operation is queued at the control unit. When control unit queueing is provided, DCWs specified by the TCCB shall be executed up to a point where the I/O resources are unavailable to execute a DCW in that TCCB (such resources would typically be in use executing I/O operations specified in other channel programs). Depending upon the control unit design, queueing of a DCW may occur after receipt of the transport-command IU or during the processing of the TCCB. I/O operations executed up to and including the DCW to be queued may involve data transfer as a normal course of DCW execution. In all cases, queueing of a DCW for the operation shall not affect ongoing operations or cause the sequential order of DCW execution to be altered.

When the queue time is zero, either no accumulation shall have occurred, queueing is not supported, a failure of the queue-time facility shall have been recognized, or an overflow condition shall have been recognized.

If a queue-time facility failure has occurred, the control unit shall send a transport-response IU with unit check status and sense information describing the failure to allow device-dependent program recovery. If a queue-time overflow condition has been recognized, unit check status shall not be presented to the channel. When a queue-time facility failure has been recognized and after unit-check status has been presented and accepted, the control unit may either:

a) disable its queue-timing facility and process subsequent I/O operations without queueing until this failure condition is corrected; or

b) continue queueing the subsequent I/O operations and present a value of zero in the queue-time field.

When the TV bit in the ES flags is zero, word 4 does not contain queue time information. When unit-check status is provided as the result of a timing-facility failure, the TV bit shall be set to zero in the ES flags.

NOTE 43 – The preferred implementation for control units that perform queueing and have detected a queueing-timing-facility-failure condition is to continue queueing subsequent I/O operations and present a value of zero in the queue-time field.

8.15.6.6.4 Device-Busy Time

When the TV bit in the ES flags is one, word 6 contains a 32-bit binary unsigned integer that indicates the device-busy time for the TCW I/O operation. The resolution of bit 31 of the device-busy time is one microsecond. The frequency offset to nominal frequency of the oscillator used to calculate the time interval shall not exceed 150 x 10^{-6}.

The device-busy time represents the time that a transport-command IU was delayed at the control unit because of a device busy condition caused by the device being locked or reserved for another operation.

When the device-busy time is zero, either no accumulation shall have occurred, a failure of the timing facility shall have been recognized, or an overflow condition shall have been recognized.

If a timing facility failure has occurred, the control unit shall send a transport-response IU with unit check status and sense information describing the failure to allow device-dependent program recovery. If a device-busy-time overflow condition has been recognized, unit check status shall not be presented to the channel. When a timing facility failure has been recognized and after unit-check status has been presented, the control unit shall continue with execution of subsequent channel programs as before; however, it shall present a device-busy time of zero (in this case, unit-check status shall not be presented after the initial indication of a timing facility failure).

When the TV bit in the ES flags is zero, word 5 does not contain device-busy time. When unit-check status is provided as the result of a timing-facility failure, the TV bit shall be set to zero in the ES flags.

8.15.6.6.5 Device-Active-Only Time

When TV bit in the ES flags is one, word 7 contains a 32-bit binary unsigned integer that indicates the device-active-only time for the TCW I/O operation. The resolution of bit 31 of the device-active-only time is one microsecond. The frequency offset to nominal frequency of the oscillator used to calculate the time interval shall not exceed 150 x 10^{-6}.

The device-active-only time represents the time that the transmission of the transport-response IU was delayed at the control unit while waiting for final DE status from the device after final CE status was recognized by the control unit for a TCW I/O operation. This value is only provided when the device status contains both CE and DE status. When CE is presented without DE status, word 6 is set to zero.

When the device-active-only time is zero, either no accumulation shall have occurred, a failure of the timing facility shall have been recognized, or an overflow condition shall have been recognized.

If a timing facility failure has occurred, the control unit shall send a transport-response IU with unit check status and sense information describing the failure to allow device-dependent program recovery. If a device-active-only time overflow condition has been recognized, unit check status shall not be presented to the channel. When a timing facility failure has been recognized and after unit-check status has been presented, the control unit shall continue with execution of subsequent channel programs as before; however, it shall present a device-active-only time of zero (in this case, unit-check status shall not be presented after the initial indication of a timing facility failure).

When the TV bit in the ES flags is zero, word 6 does not contain device-active-only time. When unit-check status is provided as the result of a timing-facility failure, the TV bit shall be set to zero in the ES flags.

8.15.6.6.6 I/O Status Additional Data

The additional-data area for I/O-status extended status may contain up to 32 bytes of additional data associated with the transport-mode I/O operation. If the ES length is greater than 32 then additional data has been provided and the amount is equal to the ES Length minus 32.

Sense data shall be provided in the additional-data area when unit-check status is included in the device status. Sense data provides information concerning unusual conditions detected in a previous

I/O operation and concerning the actual state of the device. Sense data shall provide more detailed information than the status byte and may describe reasons for the unit-check indication. It may also indicate that the device is in the not-ready state or that a drive is in the write-protected state.

Sense data shall be sent in ascending order starting with sense-data byte 0 as the first byte of the sense data. The length of sense data sent shall be from a minimum of one byte to a maximum of 32 bytes. Bits 0-5 of the first sense-data byte (sense byte 0) shall be common to all devices. The six bits shall be independent of each other and, when set to ones, shall specify the events described in table 18. The number of bytes of sense data beyond one is device dependent.

When unit-check status is not included in the device status, any additional data provided is device dependent.

8.15.6.7 I/O Exception Extended Status Format

When the ES-type code in the ES flags indicates the I/O-exception type code, the format of the extended-status shall be as shown in figure 60.

0	ES Length	ES Flags	DCW Offset	
1	DCW Residual Count			
2	Reserved			
3	Reserved			Reason Code
4	Reason Code Qualifier			
7				
8	Additional Data (when ES Length > 32)			
N				
	0	8	16	31

Figure 60 – I/O-Exception Extended Status Format

Words 0-2 are as defined in 8.15.6.1.

Reason Code (RC): Byte 3 of word 3 shall contain the reason code. The reason code shall be an 8-bit unsigned integer code that indicates the reason for the exception. The meaning of each reason code is defined in table 30. The reason-code qualifier in the extended status contain may contain additional information for each reason code.

.

Table 30 – I/O exception Reason Code (RC)

Value	Meaning
0	No information: byte 3 has no meaning.
1	TCCB integrity error: the control unit has determined that the TCCB arrived in a corrupted state. The reason code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 4 is indicated in the status-flags1 field.
2	Invalid CRC detected: invalid CRC was detected on received data. The reason-code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 4 is indicated in the status-flags1 field.
3	Incorrect TCCB length specification: the reason-code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
4	TCAH specification error: the reason-code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
5	DCW specification error: there is an error with the DCW designated by the DCW-offset field in the extended status. The reason-code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
6	Transfer-direction specification error: the command specified by the DCW designated by the DCW-offset field in the extended status specifies a direction of data transfer that disagrees with the transfer direction specified in the TCH or both the R and W bits are set to one in the TCH and bidirectional data transfer is not supported by the control unit. The reason-code qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
7	Transport-count specification error: the reason-code-qualifier (RCQ) field contains additional information. This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
8	Two I/O operations active: While an I/O operation is active at the device a second non-interrogate TCCB has been transported to the device for execution. The RCQ field has no meaning.This reason code shall be provided only when exception code 4 is indicated in the status-flags1 field.
9	One or more entries in the CRC-offset block indicate that an intermediate CRC is not at a location that is appropriate for the device and/or command being executed. The reason-code-qualifier (RCQ) field contains additional information.This reason code shall be provided only when exception code 5 is indicated in the status-flags1 field.
10-255	Reserved.

Reason Code Qualifier (RCQ): Words 1-4 shall contain the reason code qualifier. The reason code qualifier may provide additional information about the reason for the I/O exception.

When the RC field contains 1, only byte 0 of the RCQ field has meaning as defined in table 31.

Table 31 – RCQ for TCCB Integrity Error

Value	Meaning
0	No additional information.
1	Data Count Error - the amount of data transferred for the transport-command IU is not equivalent to the amount of data specified by the L1 field plus 8 in the TCH for control units that do not support bidirectional operations or is not equivalent to the amount specified by the L1 field plus either 8 or 9 for control units that do support bidirectional operations.
2	LRC error - the LRC on the transport-command IU is invalid.
3-255	Reserved.

When the RC field contains 2, only words 0-1 of the RCQ field have meaning as defined table 32.

Table 32 – RCQ for Output Data CRC Error

Word	Meaning
0	Word 0 contains the 32-bit unsigned integer offset, relative to the first byte of data received for this TCCB, of the first output-data byte in the block of data for which the invalid CRC was detected.
1	Word 1 contains the 32-bit unsigned integer offset, relative to the first byte of data received for this TCCB, of the last output-data byte in the block of data for which the invalid CRC was detected.
2-3	Reserved.

When the RC field contains 3, only byte 0 of the RCQ field has meaning as defined in table 33.

Table 33 – RCQ for Incorrect TCCB Length Specification

Value	Meaning
0	No additional information.
1	The L2 field does not specify a number of bytes that is 8 bytes greater than the number of bytes specified by the L1 field.
2	The value specified by the L2 field is less than 20 or greater than 252.
3-255	Reserved.

When the RC field contains 4, only byte 0 of the RCQ field has meaning as defined in table 34.

Table 34 – RCQ for TCAH Specification Error

Value	Meaning
0	No additional information.
1	Format control specification error: The format control specifies an unrecognized format.
2	Reserved-field specification error: a reserved field in the TCAH that is required to contain zeros contains a non-zero value.
3	Service-action-code-field specification error: The service-action-code field in the TCAH contains an unrecognized value or a value that is incorrect for the command specified by the DCW designated by the DCW-offset field in the extended status.
4-255	Reserved.

When the RC field contains 5, only byte 0 of the RCQ field has meaning as defined in table 35.

Table 35 – RCQ for DCW Specification Error

Value	Meaning
0	No additional information.
1	Reserved-field specification error: A reserved field in the DCW that is required to contain zeros contains a non-zero value.
2	Flags-field command-chaining specification error: Any of the following are true: • The chain-command flag is one and the next DCW location is within the TCA but the location is less than 8 bytes from the end of the TCA; • The chain-command flag is one and the offset of the next DCW is such that all of the next DCW is beyond the end of the TCA and a TCAX has not been specified; • The chain-command flag is one and the next DCW location is within the TCAX but the location is less than 8 bytes from the end of the TCAX. • The chain-command flag is zero and more than 3 unused bytes remain in the TCA.
3	Control-data count specification error: A DCW in the TCA specifies control data past the end of the TCA or a DCW in the TCAX specified control data past the end of the TCAX.
4	TCOB DCW or TeCOB DCW location error: The first TCOB or TeCOB DCW is not the first DCW in the TCA.
5	TCOB DCW or TeCOB DCW duplication error: More than one TCOB or TeCOB DCW is specified in the TCA and TCAX.
6	TCOB DCW or TeCOB DCW multiple-count specification error: Both the control data count and the data count are either zero or non-zero.
7	TCOB DCW or TeCOB DCW direction error: A TCOB or TeCOB DCW is specified in the TCA and the W bit in the TCH is zero.
8	TCOB DCW or TeCOB DCW chaining error: The chain-command flag in the TCOB or TeCOB DCW is zero.
9	TCOB or TeCOB count-specification error: A TCOB or TeCOB DCW has a nonzero control data count or a data count that is not a multiple of 4.
10	TTE DCW location error: Neither a TCOB DCW nor a TeCOB DCW was specified and a TTE DCW was encountered that was not the first DCW in the TCA or a TCOB DCW or TeCOB DCW was specified and the first TTE DCW encountered was not the second DCW in the TCA.
11	TTE DCW duplication error: More than one TTE DCW is specified in the TCA and TCAX.
12	TTE DCW CD-count specification error: The control-data count in the TTE DCW specifies a value that is not zero.
13	TTE DCW data-count specification error: The data count in the TTE DCW specifies a value that is less than 8 or a value that is not a multiple of 4.
14	TTE DCW direction error: A TTE DCW is specified and the W bit in the TCH is zero.
15	TTE DCW chaining error: The chain-command flag in the TTE DCW is zero.
16	TCAX specification error: A TTE DCW is specified and any of the following are true: • The TCA contains only transport-command DCWs; or • The last DCW in the TCA does not have the chain-command flag set to one.
17-255	Reserved.

When the RC field contains 6, only byte 0 of the RCQ field has meaning as defined in table 36.

Table 36 – RCQ for Transfer-Direction Specification Error

Value	Meaning
0	No additional information.
1	Read-direction specification error: The DCW specifies an input operation and the R bit in the TCH is zero.
2	Write-direction specification error: The DCW specifies an output operation and the W bit in the TCH is zero. Note - A DCW specification is recognized when a TCOB, TeCOB, or TTE DCW is specified and the W bit in the TCH is zero (see table 35).
3	Read-Write conflict: Both the R and W bits in the TCH are one and the control unit does not support bidirectional operations or the control unit supports bidirectional operations but the transport-command IU did not contain the BRDL field, or both the R and W bits are not one and the transport-command IU contains the BRDL field.
4-255	Reserved.

When the RC field contains 7, only byte 0 of the RCQ field has meaning as defined in table 37.

Table 37 – RCQ for Transport-Count Specification Error

Value	Meaning
0	No additional information.
1	Read-count specification error: for read operations, the DL field in the TCCB specifies a value that is not equivalent to the total count of data bytes specified by the DCWs in the TCA plus pad bytes and CRC; for bidirectional operations, the BRDL field in the TCCB specifies a value that is not equivalent to the total count of data bytes specified by the read DCWs in the TCA plus pad bytes and CRC.
2	Write-count specification error: The DL field in the TCCB specifies a value that is not equivalent to the total count of data bytes specified by the write DCWs in the TCA plus intermediate pad bytes, intermediate CRC, final pad bytes and final CRC bytes, and if a COB or eCOB is included in the first transport-data IU of the write data transfer, the count includes the COB or eCOB, any COB or eCOB pad bytes and the COB or eCOB CRC bytes. If a TTE DCW is present in the TCA, the count also includes the TCAX and TCAX CRC bytes.
3-255	Reserved.

When the RC field contains 9, only words 0-1 of the RCQ field have meaning as defined table 38.

Table 38 – RCQ for COB Error

Word	Meaning
0	Word 0 contains the 32-bit unsigned integer byte offset, relative to the first byte of the COB or eCOB received for this TCCB, of the first entry which contains a value that is not appropriate for the device and/or command being executed.
1-3	Reserved.

8.15.6.7.1 I/O Exception Additional Data

The additional data area for the I/O-exception type code may contain up to 32 bytes of vendor specific data. If the ES length is greater than 32 then additional data has been provided and the amount is equal to the ES Length minus 32.

8.15.6.8 Interrogate Extended Status Format

When the ES-type code in the ES flags indicates the Interrogate type code, the format of the extended status is as shown in Figure 61.

0	ES Length	ES Flags	DCW Offset	
1	DCW Residual Count			
2	Reserved			
3	Format	Flags	CU State	Device State
4	Operation State	Reserved		
5 7	State Dependent Information			
8	Device Level Identifier			
9 N	Device Dependent Information (when ES Length > 36)			
	0	8	16	31

Figure 61 – Interrogate Extended Status Format

Words 0-2 are as defined in 8.15.6.1

8.15.6.8.1 Format

Byte 0 of word 3 shall contain the format. The format is an 8-bit unsigned integer that defines the layout of the interrogate TSA. If the value of this field is not one, the contents of the interrogate extended status are meaningless.

8.15.6.8.2 Flags

Byte 1 of word 3 shall contain interrogate flags. The flags contain information about the interrogate TSA. Bits 3-7 shall be set to zero by the control unit. The meaning of each bit is defined in Figure 62.

CSV	DSV	OSV	reserved	
0	1	2	3	7

Figure 62 – Interrogate Flags

Control-unit-state valid: When bit 0 is one, the control-unit-state field contains meaningful information. When bit 0 is zero, the control-unit-state field has no meaning.

Device-state valid: When bit 1 is one, the device-state field contains meaningful information. When bit 1 is zero, the device-state field has no meaning

Operation-state valid: When bit 2 is one, the operation-state field contains meaningful information. When bit 2 is zero, the operation-state field has no meaning.

8.15.6.8.3 Control-Unit State (CS)

Byte 2 of word 3 contains an 8-bit unsigned integer that indicates a current state of the control unit for the I/O device. The meaning of each value is defined in table 39.

Table 39 – Interrogate CU State

Value	Meaning
0	Busy - the control unit is busy and the device-dependent-data field may contain additional information about the busy state.
1	Recovery - the control unit is performing a recovery process and the device-dependent data field may contain additional information about the recovery state.
2	Interrogate maximum - the control unit is executing the maximum number of interrogate operations that it supports.
3-127	Reserved.
128-255	Device-dependent meanings

8.15.6.8.4 Device State (DS)

Byte 3 of word 3 contains an 8-bit unsigned integer that indicates the state of the I/O device when the interrogate was performed. The meaning of each value is defined in table 40.

Table 40 – Interrogate Device State

Value	Meaning
0	Path-group identification - the state-dependent-information field contains information identifying a path group.
1	Long busy - the device is in a long-busy state. The meaning of long busy is device dependent and the device-dependent field may contain additional information about the long-busy state.
2	Recovery: The device is performing a recovery process and the device-dependent-data field may contain additional information about the recovery state.
3-127	Reserved.
128-255	Device-dependent meanings

8.15.6.8.5 Operation State (OS)

Byte 0 of word 4 contains an 8-bit unsigned integer that indicates whether an I/O operation is present at the device and, when present, the state of the operation. The meaning of each value is defined in table 41.

Table 41 – Operation State

Value	Meaning
0	No I/O operation is present.
1	An I/O operation is present and executing.
2	An I/O operation is present and waiting for completion of an I/O operation that was initiated by another configuration.
3	An I/O operation is present and waiting for completion of an I/O operation that was initiated for the same device extent.
4	An I/O operation is present and waiting to perform a device-dependent operation
3-127	Reserved.
128-255	Device-dependent meanings

8.15.6.8.6 State-Dependent Information

Words 5-7 contain state-dependent information. Whether this field has meaning is designated by the CS, DS, and OS fields.

The contents of this field are device dependent. When the size of the range of meaningful information in this field is less than the field size, the range of meaningful information is left justified.

8.15.6.8.7 Device-Level Identifier

Word 8 contains a device dependent token that identifies the implementation level of the device.

8.15.6.8.8 Device-Dependent Information

The device-dependent-information area for the Interrogation-type code may contain up to 28 bytes of additional data. If the ES length is greater than 36 then additional data has been provided and the amount is equal to the ES Length minus 36.

When additional data is provided, it contains device-dependent information that is defined by the contents of CS, DS and OS fields. When the size of the range of meaningful information in this field is less than the field size, the range of meaningful information is left justified.

8.15.6.9 Extended Status LRC

Extended-status LRC shall be provided when extended status is provided. The extended-status LRC, when provided, is the last word in the transport-response IU. The LRC shall be generated by the originator and recipient using a seed of hex’A5 5A A5 5A’ that is exclusive or’ed on a word by word basis over the following fields in the transport-response IU:

a) FC-SB-6 header word 0

b) FC-SB-6 header word 1 bytes 0-1 concatenated with 2 bytes of all zeros.

c) Extended Status (words 8-N)

The recipient of the transport-response IU shall compare the generated value to the extended-status LRC contained in the IU. If the value of the extended-status LRC contained in the transport-command IU equals the value generated by the recipient, a valid LRC shall be recognized; otherwise, an invalid LRC shall be recognized. If a valid LRC is recognized, the contents of the extended status shall be considered valid; otherwise, the extended status shall be considered invalid and an extended-status LRC error shall be recognized.

See Annex C for a description of the procedure and an example of the LRC calculation.

8.16 Transfer Ready IU

8.16.1 Transfer Ready Structure Overview

A transfer-ready IU shall be sent as a data descriptor FC-FS-4 information category and only by the control unit. It shall be used to request write data during a TCW I/O operation. The transfer-ready IU contains a 4-byte relative-offset field, a 4-byte maximum-burst-length field, and a 4-byte reserved field as shown in Figure 17.

A transfer-ready IU shall be sent on an open exchange. The IU keeps the exchange open and transfers sequence initiative.

A transfer-ready IU shall be sent by the control unit only during write operations. It shall be sent in response to a transport-command IU that initiates a write operation when first-transfer-ready is not disabled. Otherwise, it shall be sent in a response to a transport-data IU that did not exhaust the write data byte count for the operation and only when an abnormal condition does not exist for the operation.

8.16.2 Relative Offset

The relative-offset field contains a value specifying the relative offset for the first data byte of the requested data. The field shall have a value that is a multiple of 4 such that all data transfer shall begin on a word boundary.

The channel shall recognize a device-level protocol error if the relative offset received in a transfer-ready IU does not correspond to the relative offset maintained by the channel for the write operation.

8.16.3 Burst Length

The burst-length field contains a value indicating the maximum amount of data that can be transferred by the channel in the next transport-data IU on this transport exchange.The value in the field shall not exceed the amount of data remaining to be transferred for the write operation, shall not exceed the value specified by the DL field in the transport-command IU, and shall not be zero.

8.17 Transport Confirm IU

A transport-confirm IU shall be sent in a solicited control IU only by the channel. It shall be sent in response to a transport-response IU that does not close the transport exchange. The transport-confirm

IU provides confirmation to the control unit that a transport-response IU has been received at the channel. The transport-confirm IU shall not contain a payload and closes the exchange on which it was sent.

A transport-confirm IU shall not be sent to the channel.

9 Device-Level Functions and Protocols

This clause describes the functions and protocols necessary for the execution of an I/O operation, the exchange of control information, and device-level recovery. These functions depend on the successful completion of all FC-FS-4 and FC-LS-3 protocols.

9.1 Device-Level Operations

9.1.1 Overview of Device-Level Operations

Initiation of an I/O operation, the transfer of data, and the ending of an I/O operation are device-level operations which rely on FC-SB-6 device-level functions and protocols. The device-level functions discussed in this section have all been defined in clause 8. This section describes how these functions are used in order to perform device-level operations.

An I/O operation that is performed using the command and command information provided in a channel-command word (CCW) is referred to as a CCW I/O operation. An I/O operation that is performed using the TCCB identified by a transport-control word (TCW) is referred to as a TCW I/O operation. A TCW used to perform an I/O operation may identify an interrogate TCW that can be used to perform an interrogate operation for the TCW I/O operation (see 9.3.3.2).

A CCW channel program consists of either a single CCW or multiple CCWs that are performed sequentially in order, a process referred to as CCW chaining. A TCW channel program consists of a single TCW.

9.1.2 Channel Program Execution

A channel may concurrently execute one or more channel programs on the same or different logical paths. Each CCW channel program executed by a channel shall be for a different logical path and device and shall require a different exchange pair to be used. Each TCW channel program shall be for a different logical path and device and shall require a different transport exchange to be used.

An interrogate operation may be performed for a logical path and device when a TCW I/O operation is in progress at that logical path and device. An interrogate transport-command IU specifies a single DCW containing the interrogate command code (see 9.3.3.2). The interrogate transport-command IU is not considered part of a new or existing channel program but rather as a separate operation that is used to determine the state of a device and that does not initiate new activity at the device. The interrogate transport-command IU is sent on a new transport exchange.

The number of channel programs that may be concurrently executed by a channel or control unit is model dependent.

9.2 CCW I/O operations

9.2.1 Initiating a CCW I/O Operation

A CCW I/O operation shall be initiated with a device when the channel transfers the command from the current CCW to that device. The command shall be transferred in the command header of a command IU (see 8.7.2). CCW I/O operations are always performed in command mode.

A channel executing a CCW channel program may concurrently initiate multiple I/O operations with the same device. Each I/O operation shall consist of the transfer of a command in a command IU containing either a command DIB or a command-data DIB and, in the case of a write operation with a nonzero CCW count, the transfer of data in a data IU.

A control unit that receives one or more command IUs, each initiating an I/O operation, shall execute each command in the order in which it is received from the channel. Execution of the command first shall require acceptance of the command at the control unit. When a command is accepted, that command shall be considered current at the control unit until execution at the device is considered completed. At least the following conditions shall be satisfied in order for the command to be accepted:

a) The AS bit shall be set to one in the IUI;

b) For those commands that require the device to be installed and ready, the device address shall specify a device that is installed and ready;

c) For command chaining, the device address shall designate the same device that was designated for the command IU which initiated the channel program;

d) The chaining (CH) bit shall be set to zero for the first command of a channel program and set to one for all subsequent commands of the channel program;

e) For those devices that provide commands executed as immediate operations, the command shall be checked to determine whether a command response or status is to be sent; and

f) If the REX flag bit is set to one, the CD, DU, and SSS flags shall be set to zero in the command header.

At least the following conditions shall be satisfied in order for the control unit to return busy status to the channel.

a) The status DIB indicating device busy or control unit busy (status modifier and busy) may be sent only in response to the first command of a channel program (CH bit set to zero in the command IU);

b) A response of control unit busy shall require that the IUI of the command IU be checked. No other device-level checking shall be required; and

c) A response of device busy shall require that the IUI and device address be checked. No other device-level checking shall be required.

When the command is the first command of the CCW channel program, the channel shall consider a connection to exist when the command response or status is received. The control unit shall consider a connection to exist when the command response or status is sent.

For the first command of a CCW channel program, the channel shall send the command in a command IU with either a command DIB or command-data DIB. If the command is for a write operation and the CCW count is nonzero, a command-data DIB containing both the command and data for that command shall be used. If the command is for either a write operation with a zero CCW count or a read operation, a command DIB shall be used.

The channel shall consider the first command of a CCW channel program to be accepted when either a command response or status indicating channel-end is received.

For the first command of a CCW channel program, the channel shall consider the I/O operation to be in progress at the device when the indication that the command has been accepted by the device is received. In this case, the device shall consider the I/O operation to be in progress when the indication that the command was accepted is sent to the channel. Acceptance of the command by the control unit shall signal the start of execution of the channel program.

When the first command of a CCW channel program is handled by the control unit as a nonimmediate operation, acceptance of the command for execution shall be indicated by sending a command response to the channel. If the operation is a read, data may be sent by the control unit after the command-response IU is sent. If the operation is a write with a nonzero CCW count, data shall be sent by the channel with the command and, if necessary, in data IUs immediately following the command IU.

When the first command of a CCW channel program is handled by the control unit as an immediate operation, acceptance of the command for execution shall be indicated by sending a solicited data IU containing a status DIB containing channel-end status without busy status. Command IUs with the SYR bit set to zero and data IUs received subsequent to the command IU containing the first command of the CCW channel program, shall be discarded. If device-end status is not included with the channel-end status, the channel shall indicate that the status is accepted by sending a status-accepted IU. The chaining bit (CH bit) shall be set to one in the status-accepted IU if chaining is to be indicated by the channel on acceptance of the status; if chaining is not to be indicated, the CH bit shall be set to zero on acceptance of the status. Execution of the command shall be permitted when the command is accepted.

When device-end status is included with the channel-end status for an immediate operation, the response from the channel indicating status is accepted depends on whether or not chaining is to be indicated by the channel. If chaining is not to be indicated, the channel shall send a status-accepted IU with the CH bit set to zero. If chaining is to be indicated by the channel, then the channel shall indicate status is accepted by sending the next command in a command IU with the CH and SYR bits set to one.

When the status DIB received by the channel in response to the first command of a CCW channel program indicates that supplemental status is available, the status may be accepted as previously described, and acceptance of the supplemental status shall be indicated by the ES bit (see 9.2.3.2).

When the status received by the channel in response to the first command of a CCW channel program requests that command retry be performed, the channel shall indicate acceptance of status and its intent to perform the retry by sending either a status-accepted IU or a command IU, depending on whether device-end status is present in the status DIB and on whether the channel intends to perform the command retry. The response from the channel shall depend on the following conditions:

a) If device-end status is not present, the channel shall indicate acceptance of the status by sending a status-accepted IU. If command retry is to be performed, the CH bit shall also set to one in the status-accepted IU. From the time when the status-accepted IU is sent, the channel shall consider the rules for command chaining to be in effect (see 9.5.2). Conversely, if command retry is not to be performed, the CH bit shall be set to zero in the status-accepted IU;

b) If device-end status is present and command retry is to be performed, the channel shall indicate acceptance of the status and of the command retry by sending a command IU with the command to be retried, along with the flags and parameters associated with the command. The CH and SYR bits shall be set to one in the command IU. The CH bit shall indicate that this command is a command update as a result of command chaining and the SYR bit shall identify this command as the one being retried. From the time the command IU is sent to the device, the channel shall consider the I/O operation to be in effect at the device and the rules for command chaining in effect (see 9.5.2); and

c) If device-end status is present and command retry is not to be performed, the channel shall indicate acceptance of the status by sending a status-accepted IU. The CH bit shall be set to zero in the status-accepted IU to indicate that command retry is not to be performed.

Commands which are not executed as the first command of a CCW channel program shall not result in a command response or status being sent to the channel, unless one of the following conditions exists:

a) The CRR flag is set to one for the command; or

b) The conditions at the control unit require the sending of status (see 8.10).

The following protocols shall be used for commands which are not executed as the first command of a CCW channel program.

a) When the command is handled by the control unit as a nonimmediate operation, acceptance and execution of the command may occur immediately. If the CRR flag is set to one for the command, a command response shall be sent to the channel upon acceptance of the command. If the CRR flag is set to zero for the command, a command response shall not be sent to the channel upon acceptance of the command. If the operation is a read, data transfer may proceed immediately after sending the command-response IU, if required. If the operation is a write, the data associated with the command contained in the command-data DIB and data DIBs received may immediately be transferred to the device.

   The control unit shall consider the I/O operation to be in progress at the device when the conditions for acceptance of the command are satisfied; or

b) When the command is handled by the control unit as an immediate operation, acceptance and execution of the command may occur immediately. If the CRR flag is set to one for the command, a command response shall be sent to the channel (see 8.7.2.3). If the CRR flag is set to zero for the command, a command response shall not be sent to the channel upon acceptance of the command. At the completion of execution of the command, status may or may not be sent to the channel depending on the conditions previously described. If status is sent, and the CCW count for this command is non-zero, the control unit shall indicate that it was executed as an immediate operation by setting the LRI bit in the status flags field to a one and the residual count to the value of the CCW count.

9.2.2 Command Mode Data-Transfer Protocol

9.2.2.1 Command Mode Data-Transfer Protocol Overview

Data is transferred between the channel and the control unit as part of the execution of a CCW I/O operation. The transfer of data from the channel to the control unit is a write operation. The transfer of data from the control unit to the channel is a read operation. The operation, read or write, is determined by the current command. Read commands have bit 7 of the command byte set to zero, and write commands have bit 7 of the command byte set to one.

Some CCW read and write commands do not result in the transfer of data. These commands are executed as immediate operations, are designed to be executed without transferring data, have counts of zero in the CCW, or have data-record lengths of zero (see 9.2.1 for the protocols used when the command is executed by the control unit as an immediate operation; see 9.2.2.6 for the protocols used when the CCW count is zero).

When, for the first command of a CCW channel program with a nonzero CCW count, the data-record length at the control unit is zero but the command is not accepted as an immediate operation, a command-response IU is sent to the channel. Following transmission of the command response, if status is required to be sent to the channel, status containing channel end with or without device end is sent to the channel, the residual-count-valid bit is set to one, and the residual count field contains the CCW count received with the command. For this case, if the command is a read command, no data is transferred to the channel. If the command is a write command and data was sent for the command, the data is discarded by the control unit.

When, for a command other than the first command of a CCW channel program with a nonzero CCW count, the data record length at the control unit is zero, a command-response IU is sent to the channel only if the CRR flag is set to one for the command. If conditions require the sending of status to the channel, status is sent. In the case in which a command response is required and conditions require the sending of status, the status is sent after the command response is sent. When status is sent, the residual counts and data transfer rules for the first command of a CCW channel program apply. If conditions do not require the sending of status and chaining is to take place at the control unit, only a command response, if requested is sent; otherwise, nothing is sent to the channel.

Data is transferred from the channel to the control unit in a command-data DIB sent in a command IU or data DIB sent in a data IU. Data is transferred from the control unit to the channel only in a data DIB. See 8.8 and 8.9 for the maximum DIB sizes.

For each CCW that results in a command IU being sent, the CCW count is included in the command header. For a write operation, the CCW count is used to determine the quantity of data to expect from the channel. For a read operation, the CCW count is used to determine the quantity of data that may be sent to the channel. For both a read and write operation the CCW count is used to determine the residual count to be returned to the channel, if status is to be presented, and it is used in conjunction with the CCW control flags and command flags, to determine if chaining is to occur.

9.2.2.2 Command Mode Write Operation

A write operation in command mode is the transfer of data from the channel to the control unit as part of the execution of a command that has bit 7 set to one.

For a write operation with a nonzero CCW count, the channel shall send the command and the data for the command in a command-data DIB. If the quantity of data specified by the count for the current CCW exceeds the maximum allowable DIB size, the remaining data shall be sent in data DIBs until all of the data specified by the CCW count is sent (see 8.8 and 8.9). If all of the data specified by the CCW count is sent, the E bit shall be set to one in the last IU sent containing the CRC field. If all of the data specified by the CCW count is not sent, the EE bit shall be set to one in the last IU sent containing the CRC field.

When, during a write operation, the channel is able to send some but not all of the data for a CCW, the EE bit shall be set to one in the last IU sent. The channel, for the affected channel program, shall suspend sending IUs to the control unit until status is received. When status is received, the channel may or may not resume execution of the channel program with the control unit, depending on the contents of the status DIB and conditions at the channel.

When, during a write operation, the channel is unable to send any of the data specified for a CCW, the EE bit shall be set to one for the command-data DIB, the command header shall contain the required information for the CCW, and the DIB data field shall contain only a CRC field. If CRC generation and checking is provided, then the CRC field shall contain only the initialized value of the CRC generator in the DIB data field and the CNP bit shall be set to zero.

When, for a write operation, the quantity of data required by a device is equal to or greater than the CCW count, the resulting residual count shall be zero. If the quantity of data required by the device is less than the CCW count, the resulting residual count shall be the difference between the quantity of data required and the CCW count.

When, at the end of a write operation which is not the first command of a CCW channel program executed as an immediate operation, status including channel end is sent to the channel, the residual count calculated shall be included in the status DIB. If the quantity of data required by the device is greater than the CCW count, the LRI and RV status flag bits shall be set to one and the residual count set to zero to indicate that the device would have used more data had it been provided.

When the quantity of data received by the control unit for a write operation is less than the quantity specified by the CCW count and the EE bit is set to one in the last IU received, the I/O operation shall be executed and the data received, if any, shall be used by the device. If the quantity of data required by the device is exactly equal to the quantity of data received, the E bit shall be set to one for the status DIB sent. If the quantity of data required by the device is either less than or greater than the quantity of data received, the E bit shall not be set to one for the status DIB sent; if a valid CRC value is received and CRC checking is provided, CRC shall be checked on the entire amount of data received.

9.2.2.3 Command Mode Read Operation

A read operation in command mode is the transfer of data from the control unit to the channel as part of the execution of a command that has bit 7 set to zero.

For a read operation the channel shall send the command to the control unit in a command DIB in a command IU. The CCW count and CCW control flags for the command shall be included in the command header.

When, for a read operation, the quantity of data sent to the channel exactly equals the count for the current CCW, the resulting residual count shall be zero and the E bit shall be set to one in the last data IU sent for that CCW. If the quantity of data sent to the channel is less than the count for the current CCW, the resulting residual count shall be the difference between the quantity of data sent to the channel and the CCW count. The E bit shall be set to zero and the EE bit shall be set to one in the last data IU sent for that CCW. If more data would have been transferred to the channel had a larger count been provided or data chaining indicated, the resulting residual count shall be zero and the E bit shall be set to one in the last data IU sent for that CCW.

When, at the end of a read operation which is not the first command of a CCW channel program executed as an immediate operation, status including channel end is sent to the channel, the residual count calculated shall be included in the status DIB. If the quantity of data sent to the channel by the device, is exactly equal to the CCW count but the device would have sent more data had a larger CCW count been provided, the LRI and RV status flag bits shall be set to one and the residual count is set to zero.

9.2.2.4 Data Chaining

When there are successive CCWs to be executed by the channel for a single I/O operation, the execution of these CCWs is referred to as data chaining. When each CCW associated with a CCW channel program becomes the current CCW being executed, the channel shall indicate to the control unit the initiation of a data chaining operation by transferring a command-IU containing the chain-data flag, CCW count, and other flags associated with the CCW. Subsequent IUs containing data DIBs may also be transferred, either by the channel or control unit, depending upon whether input or output was specified in the command IU initiating the I/O operation. When execution of the current CCW is completed and the chain-data flag is set to one, data chaining shall take place, provided no abnormal conditions are detected and all other conditions for data chaining are satisfied. Data chaining may occur only when the CD bit is set to one for the current CCW. When data chaining takes place, the command IU shall be used to update the flags, CCW number, token, and count held at the control unit, the chaining bit (CH bit) of the device header flag field and the data-chaining-update (DU) flag of the command-flag field shall both be set to ones.

The device shall ensure that data chaining is occurring at the proper times by recognizing a data-chaining condition. The data-chaining condition shall be recognized for each device and shall be used to verify that successive CCWs are being executed by the channel. The data-chaining condition shall be set whenever the device accepts a command IU and the chain-data bit is set to one in the command-flag field and when all other conditions for data chaining are satisfied. The data-chaining condition shall be reset whenever a control IU is received and system reset, selective reset, cancel or stack status is performed. The data-chaining condition shall also be reset whenever a command IU is received and the chain-data bit is set to one and an end-of record condition is recognized or when the chain-data bit is set to zero, or status containing channel end, with or without device end, has been accepted by the channel.

If a command IU is received with the DU flag bit and the CH bit set to ones, and if the last byte of data has been transferred for the current CCW, then the command flags, CCW number, token, and count from the command IU just received shall be accepted and become current as the new CCW assumes control of the I/O operation.

The chain-data bit in the CCW-flag field of the command IU shall indicate that there is a subsequent count and command-flag update for the device that is to immediately follow the execution of the current command IU, provided that no abnormal conditions are encountered.

In some cases while data chaining, the channel may request the transfer of status when the last CCW of a data chain is executed. The channel shall do this by setting the SSS bit to one signalling the control unit to provide ending status in an IU containing a status DIB when the I/O operation has ended. In this case, if command chaining was set to one in the last CCW of the data chain, then normal command chaining would occur after the channel received the IU containing the status DIB. If data chaining is ended and no further chaining is specified, ending status shall be sent to the channel.

When data-chaining occurs at the channel during a write operation, a command IU with a command-data DIB containing the CCW number, the CCW count and required flags for the next CCW, a valid token, and data shall be sent to the control unit. If the quantity of data specified for the CCW exceeds the quantity of data that is allowed to be sent in a command-data DIB, one or more data DIBs may be sent. If the quantity of data sent for the CCW is exactly equal to the count in the CCW, the E bit shall be set to one in the last IU sent for the CCW. Data chaining shall continue until all of the data is transferred for the write operation or until the data chaining condition is not set at the end of a CCW. If the quantity of data sent for the CCW is less than the count in the CCW, the EE bit shall be set to one in the last IU sent for the CCW. For this case, data chaining, if indicated for the CCW, shall not be performed and execution of the channel program shall be suspended until status is received from the control unit. If the quantity of data used by the device is less than the total count provided by the channel, then a status DIB shall be sent; the residual count field shall indicate the amount of data which was not used. If the quantity of data required by the device is greater than the total count provided and the EE bit is not set, then status, if sent, shall have the LRI and RV bits in the status-flag field set to one and the residual count set to zero.

When, at the control unit for a write operation, data chaining is indicated in the current CCW and when the last byte of data for the CCW has been successfully received, data chaining occurs at the control unit and the next CCW shall go into effect. That is, the contents of the next command IU which would have the CH bit set to one and a command DIB containing the DU bit set to one shall become current and go into effect for the write operation. It should be noted that conditions may exist precluding data chaining from occurring; however, this shall not preclude control of the I/O operation from being transferred to the next CCW as it becomes current. For any status that is sent, including status sent in situations where no data has been successfully received for the current CCW, the CCW number and the status shall pertain to the current CCW.

When, at the channel for a read operation, data chaining is indicated in the current CCW, a command IU with a command DIB containing the CCW number, the CCW count and required flags for the next CCW, and a valid token shall be sent to the control unit. The channel shall continue to do this, provided sufficient IU pacing credit is available, for each CCW for which data chaining was specified until the command IUs for all of the CCWs for data chaining have been sent (see 9.2.2.5). For a read operation, the sending of the command IUs indicating data chaining does not constitute execution of the command or CCWs; the operation and data chaining shall only be considered to have been initiated. There is an indefinite period of time from the initiation of a read operation with data chaining and when data for that operation arrives at the channel. When a data IU is received by the channel, the CCW number shall indicate the CCW for which this data is being sent. At that point in time, the CCW identified shall become current at the channel and execution of the CCW shall be considered to be occurring. As each successive data IU is received for a read operation, the channel shall ensure that the CCW number specified corresponds to the appropriate CCW. The quantity of data received for a CCW shall exactly equal the CCW count for that CCW in order for data chaining to occur at the channel. If, before all of the data for a CCW is received, the next data IU received from the control unit specifies a different CCW than the current CCW for the read operation, a device-level protocol error shall be recognized. If either status or a data IU with the EE bit set to one is received, data chaining shall be ended for the current read operation.

When, at the control unit for a read operation, the quantity of data to be sent to the channel exceeds the CCW count in effect from the command IU currently being executed, the E bit shall be set to one in the last data IU sent for the CCW and if the data chaining condition is set, data chaining shall occur at the control unit when the next command IU is received and the contents of that command IU have the CH bit set to one with a command DIB containing the DU flag set to one. When data chaining occurs at the control unit, the next CCW received shall become current and go into effect for the read operation. It should be noted that conditions may exist precluding data chaining from occurring; however, this shall not prevent control of the I/O operation from being transferred to the next CCW as it becomes current. The remaining data shall be sent in data IUs under the control of the new CCW. The CCW number from the current command IU shall be used in each data DIB sent, and the CCW count shall determine the maximum quantity of data that may be sent. For any status that is generated, including status sent in situations where no data DIBs have been sent for the current CCW, the CCW number and the status shall pertain to the current CCW.

When all conditions for data chaining are satisfied, data chaining shall continue at the control unit until all of the data is transferred for the read operation or at the completion of the current CCW, the data chain condition is not set. If the quantity of data transferred is equal to the CCW count and if all of the data has been transferred for the read operation, data chaining shall not occur even though the data chaining condition is set; however, control of the I/O operation shall be passed to the next CCW, if any, and status shall be sent to the channel. If all of the data has not been transferred to the channel for the read operation and the data chaining condition is not set, data chaining shall not occur and a status DIB, if sent, shall contain the LRI and RV bits in the status-flags field set to one and the residual count set to zero.

When the data-chaining condition is set and the control unit initiates the ending of the I/O operation by transferring channel-end status to the channel, with or without device-end status, and if the quantity of data transferred is less than the CCW count, then the data-chaining condition shall be reset, control shall not be passed to the next CCW, if any, and the CCW number and status shall pertain to the current CCW.

Data chaining shall occur whenever the data-chaining condition is set, and the chaining (CH) bit in the device-header flag field and the data-chaining-update flag (DU flag) are both set to ones in the command IU.

If the data-chaining condition is set in the control unit and if a command IU is recognized with the chaining bit (CH bit) or the data-chaining-update flag (DU flag) set to zero, then a device-level protocol error shall be detected. If the data-chaining and command-chaining conditions are both not set and if a command IU is received with either the DU flag set to one or the CH bit set to one, then a device-level protocol error shall be detected. The CH bit and DU flag shall be tested before the data-chaining condition is changed to conform to the new value of the chain-data bit in the command-flag field.

If the data-chaining condition is set, the control unit shall ensure that the path to the device remains available when the current count is exhausted, until the next command IU is recognized or until the data-chaining condition is reset.

A channel may minimize the number of data chain updates for successive data chain CCWs. This is accomplished by adding the counts from successive CCWs and presenting only the sum of these counts to the control unit in a single command IU. When this is done, the original integrity of the channel program shall be maintained by ensuring that PCI and command chaining conditions occur at the appropriate points within the channel program.

9.2.2.5 IU Pacing

Each channel provides an IU pacing credit which is initialized at either the start of each CCW channel program or reconnection to continue execution of a CCW channel program. The IU pacing credit is the maximum number of IUs that a channel may send on a given outbound exchange before it receives a command-response IU, which was sent because the CRR bit was set to one, on the existing inbound exchange.

A channel may operate in the default IU pacing mode or in the persistent IU pacing mode. A channel operates in persistent IU pacing mode on each logical path for which both the channel and control unit indicate support for the persistent IU pacing optional feature and on logical paths for which both the channel and control unit have indicated support for the concurrent enablement of persistent IU pacing function. On any logical path for which persistent IU pacing mode is not supported, a channel operates in the default IU pacing mode. See 6.4.2 for additional information about optional features and 6.3.9.2 for additional information about the concurrent enablement of persistent IU pacing function. When a channel is operating in default IU pacing mode, the IU pacing credit in effect at the start of a channel program shall be set to a model-dependent value no greater than the default value of 16. When a channel is operating in persistent IU pacing mode, the IU pacing credit is initialized with the current pacing credit in effect for the logical path.

When a control unit is operating in default IU pacing mode, the control unit may request that the IU pacing credit be increased by the channel at the start of a CCW channel program or at each time the control unit reconnects with device-end status. At the start of a CCW channel program, the control unit may request that the IU pacing credit be increased by providing an IU pacing parameter in either the command response or status sent in response to the first command of a CCW channel program; when reconnecting with device-end status, the control unit may request that the IU pacing credit be increased by providing an IU pacing parameter in the status DIB. A modified pacing credit shall remain in effect for the duration of the outbound exchange. See 8.11.2.4 and 8.10.3.4 for a description of how the control unit provides the IU pacing parameter.

When a control unit is operating in persistent IU pacing mode, the control unit may request that the IU pacing credit for the logical path be modified. Any modification of the IU pacing credit shall take effect at the start of the next channel program on the logical path, and shall persist until changed again by the control unit, or until a system reset is performed for the logical path. The control unit may request that the IU pacing credit be modified by providing an IU pacing parameter in either the command response sent in response to the first command of a channel program or any status sent with the residual-count valid (RV) bit set to zero and the persistent IU pacing parameter valid bit set to one. See 8.11.2.4 and 8.10.3.4 for a description of how the control unit provides the IU pacing parameter.

A control unit may increase, decrease, or reset the pacing credit to the default value. A pacing parameter value of zero shall have the effect of resetting the credit to the default value of 16. In order to avoid resetting the pacing count to the default, the control unit shall retain its desired setting and include this value in the pacing parameter for all Command Response IUs and Status IUs in which the pacing parameter is valid. This shall be done for all device operations on the same logical path.

If the control unit sets the IU pacing parameter to a value less than or equal to the default value, the channel shall not increase the IU pacing credit above the default value. If the control unit sets the IU pacing parameter to a value greater than the default value, then the channel may increase the IU pacing credit by any amount up to the value indicated by the IU pacing parameter.

At the start of a CCW channel program or at each reconnection, the channel shall send a model dependent number of IUs to the control unit. The number of IUs sent shall not exceed the IU pacing credit value. Prior to or at the last command IU sent, the channel shall request a command response to be returned by setting the CRR bit in a command or command-data DIB. The selection of the command or command-data DIB for the setting of the CRR bit shall be such that the remaining IU pacing credit (that is, the number of additional IUs the channel is allowed to send before it receives a command-response IU) does not prevent the transmission of all of the IUs for a CCW. For example, if the channel has not set the CRR bit since the command or command-data DIB for which the last command response was received and remaining IU pacing credit is less than the number of IUs required to transfer all of the data indicated by the CCW count field in a command-data DIB, then the CRR bit shall be set to a one in the command-data DIB; otherwise the channel shall be unable to proceed with the channel program.

When a command response is received, it shall indicate which CCW is currently being executed and, therefore, the number of IUs that have been processed since the start of the channel program or since the IU for which the previous command response was received. Upon receipt of the command response, the channel is then permitted to send an additional number of IUs beyond the current remaining credit equal to the number of IUs indicated as having been processed.

When a control unit sends a data IU containing a status DIB, the control unit shall discard all command IUs with the SYR bit set to zero and data IUs which are received subsequent to the IU for which the status was sent (see 8.10). When a data IU containing a status DIB is received, the channel sets its remaining IU pacing credit to a value equal to the IU pacing credit for the exchange; the number of IUs a channel is then permitted to send, including and subsequent to the IU sent in response to the status, is equal to the IU pacing credit.

When an IU which closes the inbound exchange is received, a channel is allowed to respond to the IU without regard to IU pacing credit.

NOTE 44 – Care should be exercised when selecting the command or command-data DIB in which to set the CRR bit. Setting the CRR bit too early results in an excessive number of command-response IUs; setting the CRR bit too late may result in interrupting the flow of command and data IUs on the outbound exchange until a command-response is received. In order to avoid this and as a “rule of thumb” the CRR bit should be set in a command IU as close as possible to the point of sending IUs where half of the IU pacing credit value remains.

NOTE 45 – An IU pacing credit higher than 16 is recommended for link speeds above 1Gbits/sec. For distances of up to 100 KM, the IU pacing credit should double as the link speed doubles above 1Gbits/sec.

NOTE 46 – The intent of the IU pacing function is to prevent data-intensive channel programs such as a CCW channel program performing a data archive operation from unfairly utilizing all of the resources of a control unit. IU pacing is not intended to be a means of flow control.

NOTE 47 – A control unit may return control-unit busy status if resource limitations prevent the control unit from accepting all of the IUs sent by a channel during the initiation of a new exchange pair.

NOTE 48 – The IU pacing function is defined only for transmissions from the channel to the control unit. It is not required for control unit to channel transmissions because resources are already allocated in the channel prior to execution of the channel program.

NOTE 49 – A preferred channel implementation makes use of the IU pacing parameter provided by the control unit.

NOTE 50 – A preferred control unit implementation provides a means by which the IU pacing parameter may be modified based on operating conditions.

9.2.2.6 Zero CCW Count

When a CCW is fetched containing a count field of zeros, the CCW count field of the command IU is likewise set to zeros.

When a command with a CCW count of zero is executed, the type of response sent to the channel, if any, and whether or not command chaining, if indicated, occurs depend on how the command is executed and the flag bits in the command header.

If the first command of a CCW channel program is executed as an immediate operation, a data IU containing a status DIB shall be sent. The LRI and RV bits shall be set to zero. The E-bit shall be ignored for this case. Command chaining, if indicated, shall be initiated by the channel when device-end status is received and the conditions for command chaining at the channel are satisfied.

If the first command of a CCW channel program is executed as a nonimmediate operation, a command-response IU shall be sent. Execution of the command may proceed after the command has been accepted and the command response sent; chaining to the next CCW may proceed if all of the conditions for chaining are satisfied (see 9.2.4).

If other than the first command of a CCW channel program is executed, the CRR bit and SSS bit in the command header are both set to zero, and all of the conditions for chaining are satisfied, then execution of the command may proceed, no response shall be sent, and chaining to the next CCW may occur.

If other than the first command of a CCW channel program is executed and the CRR bit and the SSS bit are both set to one, a command-response IU followed by a data IU containing a status DIB shall be sent. In the status DIB, the E bit in the IU header shall be set to zero by the sender and ignored by the recipient, and the RV bit in the status-flags field shall be set to one for a residual count of zero. If the command would have been executed as a nonimmediate operation had a count other than zero been provided, the LRI status flag bit shall be set to one and the residual count is set to zero. Command chaining, if indicated, shall be initiated by the channel when device-end status is received and the conditions for command chaining at the channel are satisfied (see 9.2.4).

If other than the first command of a CCW channel program is executed and the CRR bit is set to one and the SSS bit is set to zero, a command-response IU shall be sent. Command chaining, if indicated, occurs at the control unit and status shall not be sent to the channel.

If other than the first command of a CCW channel program is executed and the CRR bit is set to zero and the SSS bit is set to one, a command response shall not be sent but a data IU containing a status DIB shall be sent. In the status DIB, the E bit in the IU header shall be set to zero by the sender and ignored by the recipient, and the RV bit in the status-flags field shall be set to one for residual count of zero. If the command would have been executed as a nonimmediate operation had a count other than zero been provided, the LRI status flag bit shall be set to one and the residual count shall be set to zero. Command chaining, if indicated, shall be initiated by the channel when device-end status is received and the conditions for command chaining at the channel are satisfied (see 9.2.4).

The presence of a zero count in the command IU shall not be used to determine whether a command should be executed as an immediate operation, executed as a nonimmediate operation, or rejected by returning unit-check status. For example, when a command-response IU would be returned for the first command of the channel program if the CCW count were nonzero, a command response shall also be returned when the count is zero. This is to ensure that compatibility is maintained among the Fibre Channel interface, SBCON and the parallel-I/O interface.

9.2.3 Ending a CCW I/O Operation

9.2.3.1 General Rules for ending a CCW I/O Operation

The ending of a CCW I/O operation is either channel initiated or control-unit initiated. The channel may initiate the ending of an I/O operation as the result of an abnormal condition or a non-error condition. The control unit may initiate the ending of a CCW I/O operation as the result of the completion of the execution of the command or the transfer of all data associated with the command or as the result of an abnormal condition detected during the execution of the command.

When the channel initiates the ending of a CCW I/O operation, it shall send a control IU indicating one of the following control functions:

a) cancel, (see 9.4.3),

b) selective reset, (see 9.4.5), or

c) system reset (see 9.4.4).

When conditions require the ending of a CCW I/O operation with the channel, status in a status DIB shall be sent and all command IUs with the SYR bit set to zero and data IUs received subsequent to the current command shall be discarded.

When conditions permit the ending of a CCW I/O operation without the transfer of status to the channel, chaining shall occur with the start of execution of the next command (see 8.10).

In the absence of errors, when the current command is executed as a nonimmediate operation, the control unit shall initiate the ending of the CCW I/O operation when one of the following conditions is satisfied:

a) All of the data specified by the CCW count has been transferred and the data-chaining condition is not set; or

b) The data required by the device for the command has been transferred.

When the current command is executed as an immediate operation, the control unit shall initiate the ending of the I/O operation when the conditions for the device allow channel-end status to be generated.

Ending a CCW I/O Operation without Transferring Status: When execution of the current command reaches completion with the device and conditions do not require status to be presented to the channel, the I/O operation shall be considered to have ended and execution of the next command in the chain may proceed.

Ending a CCW I/O Operation with Transferring Status: When the control unit ends execution of the current command, including the transfer of data, and conditions require status to be transferred to the channel, a status DIB indicating channel end with or without device end shall be sent along with the appropriate status flags and residual count. If the operation is ended because of an abnormal condition, the appropriate status for this condition is included. If device-end status is included, the operation shall be considered ended by the device when this status is accepted by the channel. If device-end status is not included, the I/O operation shall be considered ended by the device when the control unit later transfers device-end status and receives acknowledgment that the status has been accepted by the channel. If unit-check status is included, the control unit may optionally present supplemental status along with this status, and indicate this by setting the ES bit to one in the IU header.

If the control unit is connected when it is ready to transfer status in order to initiate the ending of the I/O operation with the channel, a status DIB with channel end with or without device-end status shall be sent (see 8.2.2 for information on when the inbound exchange is closed with the sending of the status DIB).

If the control unit is not connected when it is ready to transfer status in order to complete the ending of the I/O operation with the channel, a data IU containing a status DIB indicating device-end status shall be sent. The inbound exchange shall be left open.

9.2.3.2 CCW Supplemental Status

Supplemental status provides sense information in a status DIB to describe conditions at the control unit or device for which status information containing unit check is being provided.

The control unit may optionally transfer supplemental status along with unit-check status (but not retry status). The presence of the supplemental status shall be indicated by the ES bit being set to a one in the IU header of the status DIB containing unit-check status. The channel shall indicate it has accepted the supplemental status along with the status by setting the ES bit to a one in the status-accepted IU. If the channel has accepted the status but not accepted the supplemental status, the ES bit shall be set to a zero in the status-accepted IU. In this case, the supplemental status shall be held by the device until retrieved by the channel with sense information by the appropriate sense command, or cleared.

The ability to accept supplemental status is determined by conditions present at the channel and the design of the channel. Whether or not supplemental status is included in a status DIB sent to the channel depends on the status information being transferred and conditions at the control unit, which are model dependent.

NOTE 51 – The preferred implementation is for a control unit to provide supplemental status in the status DIB whenever permitted.

NOTE 52 – Supplemental status is presently defined only for unit-check status (but not retry status).

9.2.3.3 CCW Residual Count

The control unit shall determine the residual count by calculating the difference between the CCW count and the actual number of bytes either written to a device or read from a device and sent to the channel. The value of the residual count is dependent on the CCW count, the number of bytes transferred between the channel and control unit, and the number of bytes actually read from or written to the device.

For a write operation the channel may transfer more data to the control unit than the device requires, but never more data than specified by the CCW count. If the channel transfers all of the data specified by the CCW count and all of the data is used by the device, the residual count shall be equal to zero for this command. If a write command with a nonzero CCW count is executed as an immediate I/O operation, the residual count shall be equal to the CCW count.

When, for a write operation, the device requires less data than specified by the CCW count, the residual count shall be equal to the difference between the CCW count and the actual number of bytes of data used. Alternatively, if the device requires more data than specified by the CCW count, the residual count shall be zero for this command.

For the case where, for a write operation, the channel transfers less data than specified by the CCW count (see 8.11.2.3 and 9.2.2.2), the residual count shall still be determined by calculating the difference between the CCW count and the actual number of bytes used by the device.

For a read operation only a quantity of data equal to or less than the CCW count shall be transferred to the channel. The residual count shall be the difference between the CCW count and the actual number of bytes sent. If all of the data specified by the CCW count is transferred, the residual count shall be equal to zero for this command. If a read command with a nonzero CCW count is executed as an immediate I/O operation, the residual count shall be equal to the CCW count.

When, for a read operation, less data than specified by the CCW count is transferred to the channel, the residual count shall be equal to the difference between the CCW count and the actual number of bytes sent. If, for the read operation, all of the data specified by the CCW count is sent but more data would have been sent to the channel had a larger count been specified, the residual count shall be equal to zero.

Note that there are only three cases in which the residual count is greater than zero:

a) when the record length is less than the CCW count;

b) when, during a write operation, the channel is unable to send all of the data for a CCW and has set the EE bit in the last IU sent; and

c) when a command with a non-zero CCW count is executed as an immediate operation.

Table 14 shows the valid combinations for the LRI and RV bits, the residual count, and channel-end status. Also, note that for all cases in which a valid residual count is presented, channel-end status shall also be presented.

The following abnormal situations may require additional flags to be set in the device-header-flags field or may prevent including a valid residual count in a status DIB:

a) The E bit shall be set to one in the device-header-flags field for a status DIB only for a write operation involving an early end indication for which the amount of data received from the channel and written to the device exactly equals the record length;

b) If an abnormal condition prevents determining the exact amount of data either read from or written to a device, a valid residual count shall not be calculated. In this case, the RV bit shall be set to zero in the status DIB. Unit-check status shall be indicated along with channel-end status. Command retry may be requested. See the definition of the CR bit in 9.5.2. If the channel accepts the command retry request, the retry proceeds normally. If command retry is not honored, the channel shall terminate the I/O operation;

c) If, for other than the first command of a CCW channel program, the channel receives a status DIB with the channel-end status bit set to one, the unit-check status bit set to zero, and the residual count valid bit set to zero, a device-level protocol error shall be recognized; or

d) If during a read operation, an abnormal condition prevents determining the exact amount of data transferred to the channel, then a valid residual count shall not be calculated. In this case, the RV bit shall be set to zero and unit check along with channel-end status is indicated. Command retry may also be requested.

When, during data chaining, the next CCW takes effect, a new residual count shall be started for the data transfer associated with this CCW. If status is generated and sent to the channel, the residual count shall be included, provided that a valid residual count is calculated.

9.2.4 CCW Command Chaining

When there are successive I/O operations to be executed by a channel and a control unit for a single channel program, the execution of these successive I/O operations is referred to as command chaining. When an I/O operation is initiated by the channel, it shall use a command IU to transfer the command, CCW control flags, CCW count, and command flags associated with the current CCW to the control unit. In this manner, the control unit is informed whether or not command chaining is specified for the current CCW. When an I/O operation is completed and the chain-command flag is set to one and the chain-data flag is set to zero, command chaining shall be permitted, provided no abnormal conditions are detected and all other conditions for chaining are satisfied. When command chaining takes place, the next command and associated flags and count shall become current and the next I/O operation shall be considered to have started. The next command and associated flags and count shall be found in the next command IU received which shall have the chaining bit (CH bit) set to one and the data-chaining-update flag (DU command flag) set to zero.

The control unit shall ensure that command chaining is occurring at the proper times by recognizing a command-chaining condition. The command-chaining condition shall be recognized for each device. The command-chaining condition shall be set as a result of command chaining or as a result of command retry. During command chaining, the command chaining condition shall be set whenever a command with the CC flag set to one and the CD flag set to zero in the CCW-flag field of a command IU is accepted. The command-chaining condition shall be set for command retry when the CH bit is set to one in the IU used by the channel to accept the status that requests a command retry (see 9.5.2).

When the command chaining condition is set and the conditions for not sending status at the completion of execution of the current command are all met, command chaining shall be performed by starting execution of the command in the next command IU which shall have the CH bit set to one and the DU command flag bit set to zero (see 8.10 and Annex D for the conditions when status is sent and when chaining occurs at the control unit). If the command chaining condition is set but the channel has requested that status be sent at the completion of execution of the current command, status shall be sent to initiate command chaining with the channel. In this case command chaining shall not occur until the channel receives status with device end indicated and the channel sends a command IU for the next command to be executed.

The command-chaining condition shall be reset whenever (1) system reset or selective reset is performed, (2) the device receives a cancel or stack-status in a control IU, (3) status containing channel end but not device end for an I/O operation has been accepted by the channel and the chaining bit is set to zero in the status-accepted IU, (4) status containing device end for an I/O operation has been accepted by the channel using a status-accepted IU, (5) an address-exception condition is recognized and a control IU with a device-level-exception indication is sent, or (6) the channel accepts a valid status DIB with the AS bit set to one and the status contains unit check (except when command retry is requested and the retry requested is to be honored by the channel), unit exception, busy, attention, status modifier without any other bits set, or control-unit end, provided it is not control-unit end alone. A valid status DIB for which the AS bit set to zero shall not suppress command chaining.

Command chaining shall occur whenever the command-chaining condition is set at the device, and the CH bit and the DU bit are set to one and zero, respectively.

If the command-chaining condition is set at the device, the control unit shall ensure that the path to the device remains available when device-end status is presented until the next command is recognized or until the command-chaining condition is reset. If the command-chaining condition is set at the device and if a command IU is recognized with the CH bit set to zero or the DU flag set to one, a device-level protocol error shall be detected. If the command-chaining and the data-chaining conditions are both not set and if a command IU is recognized with the CH bit set to one, then a device-level protocol error shall be detected.

9.2.5 Priority

If support for the use of Priority is indicated (i.e. both the receiving F_Port and N_Port indicate Priority in the login service parameters) (see FC-FS-4) and Priority is in use, FC-SB-6 has the following requirements on the setting of the Priority field for command mode operations:

a) The channel shall set the same Priority value in all frames of command-mode IUs sent to a control unit for the duration of a given outbound exchange and all frames of any new outbound exchange opened as a result of any reconnection that is required to continue execution of a channel program

b) All frames of any IU sent by a control unit on an inbound exchange that is opened in response to an IU sent by the channel shall either reflect back the Priority value used by the channel in the request (i.e., in the first frame of the outbound exchange of the exchange pair) or shall not use Priority.

c) A device that has disconnected from a logical path during the execution of a chain of commands, and reconnects to continue the execution of the chain of commands shall either use the same Priority value on the reconnection that was in use on the inbound exchange prior to the disconnection or shall not use Priority. If the CU did not use Priority, then if it receives a command IU with the SYR bit set (indicating the continuation of command chaining) in response to the status IU, it may use the priority of the command IU for subsequent IUs sent on that exchange.

d) If the control unit opens an inbound exchange by sending an unsolicited data IU containing a status DIB for a purpose other than reconnection, the priority value used for the status IU is model-dependent. The priority value used by the control unit in this IU shall also be used in all subsequent IUs sent for that exchange. In this case the control unit shall not adopt the priority of a command IU that may have been sent by the channel and ‘ships passed’ with the unsolicited data IU for subsequent IUs sent on that exchange in response to the command received.

9.3 TCW I/O Operations

9.3.1 Initiating a TCW I/O operation

A TCW I/O operation shall be initiated with a device when the channel transfers the transport-command-control block (TCCB) and associated control information for the current TCW to that device. The TCCB shall be transferred in a transport-command IU (see 8.13). TCW I/O operations are always performed in transport mode.

Execution of a TCCB shall require acceptance of the transport-command IU at the control unit. When a transport-command IU is accepted, the TCCB in the IU shall be considered current at the control unit until execution of the TCCB at the control unit is considered completed. When the transport-command IU is not accepted because of an error condition, a transport-response IU is returned to the channel with the initial status flag set to one to indicate that the error occurred prior to initiating execution of the TCCB at the device. The control unit shall provide error information in the status and I/O extended status in the transport-response IU to identify the transport-command IU error.

If a control unit that supports process login receives a transport-command IU from a channel but the control unit does not have a process login in effect with the channel or has a process login in effect with the channel but does not support transport mode, the control unit shall recognize a transport operation error (see 10.8).

The following conditions shall be satisfied in order for the transport-command IU to be accepted at the control unit:

1) The transport-command IU shall satisfy the following integrity checks (see 10.9.2):

    a) For control units that do not support bidirectional operations, the L1 field in the TCH plus 8 shall specify an amount of data that is equal to the amount of data received by the control unit for the transport-command IU, otherwise, a transport-command IU integrity error due to a data count error shall be recognized;

    b) For control units that support bidirectional operations, the L1 field in the TCH plus either 8 or 9 shall specify an amount of data that is equal to the amount of data received by the control unit for the transport-command IU, otherwise, a transport-command IU integrity error due to a data count error shall be recognized; and

    c) The LRC field in the TCCB shall be valid, otherwise a transport-command IU integrity error due to invalid LRC error shall be recognized.

2) The specified logical path shall be established, otherwise a logical-path-not-established error (see 10.6) shall be recognized;

3) For those commands that require the device to be installed and ready, the device address shall specify a device that is installed and ready, otherwise an address-exception condition shall be recognized (see 9.5.4.2);

4) The R and W bits shall not both be set to one in the TCH if bidirectional data transfer is not supported by the control unit or the control unit supports bidirectional operations but the transport-command IU does not contain the BRDL field; otherwise a TCH content error (10.9.3) shall be recognized;

5) The TCCB in the transport-command IU shall satisfy all of the following conditions, otherwise, a TCCB content error (10.9.4) shall be recognized:

   a) The L2 field shall specify a length that is exactly 8 bytes greater than the L1 field and contain a value that is at least 20 bytes and not more than 252 bytes;

   b) Byte 1 of word 0 of the TCAH shall be zero;

   c) The format control field in the TCAH shall be equal to hex'7F';

   d) The service-action code in the TCAH shall contain a valid value (see 8.13.4.4);

   e) If the R and W bits are both set to zero, the data-length (DL) field shall be zero; and

6) If another TCW I/O operation is in progress for the logical path and device address specified in the transport-command IU, the service-action code shall specify hex'1FFF' and the command code in the first DCW of the TCA shall specify the interrogate command code, otherwise a non-interrogate-second-operation error shall be recognized (10.9.5).

A response of control-unit busy shall require that a transport-command IU integrity error has not been recognized and that the specified logical path be an established logical path. No TCH or TCCB content checking is required.

A response of device busy shall require that a transport-command IU integrity error has not been recognized, that the specified logical path be an established logical path and that the device address specify a device that is installed and ready. No TCH or TCCB content checking is required.

The channel shall consider a connection to exist when the transport-command IU is sent. The control unit shall consider a connection to exist when the transport-command IU is accepted.

The channel is not aware of whether the control unit has accepted the TCCB or of the progress of the I/O operation at the device until the I/O operation is terminated by the control unit with a transport-response IU. If the channel does not receive a transport-response IU within TC_TOV, the channel recognizes a transport command timeout (see 10.2.6).

9.3.2 Transport Mode Data Transfer

9.3.2.1 Transport Mode Data Transfer Overview

Data is transferred between the channel and the control unit as part of the execution of a TCW I/O operation that specifies a read, write or bidirectional operation. The transfer of data from the channel to the control unit is a write data transfer. The transfer of data from the control unit to the channel is a read data transfer. A read operation performs only a read data transfer, a write operation performs only a write data transfer and a bidirectional operation may perform both a read and write data transfer.

The data transfer is determined by the R and W bits in the TCH. Read data transfer is performed when the R bit is set to one and write data transfer is performed when the W bit set to one. Bidirectional data transfer, if supported, is performed when both the R and W bits are set to one. If neither the R or W bit is set to one, then the operation does not perform any data transfer.

Data is transferred in a transport-data IU. The maximum amount of data sent in a transport-data IU is dependent on whether the transport-data IU is for or a write transfer or read transfer as described in 9.3.2.2 and 9.3.2.3.

For read and write operations, the DL field in the TCCB specifies the amount of data to be transferred during an operation. The amount includes all required pad and CRC bytes (see 8.13.7).

For bidirectional operations, the DL field in the TCCB specifies the amount of data to be transferred for the write data transfer portion of the operation and the BRDL field specifies the amount of data to be transferred for the read data transfer portion of the operation. The amounts include all required pad and CRC bytes (see 8.13.7 and 8.13.8).

9.3.2.2 Transport Mode Write Data Transfer

A write data transfer is the transfer of data from the channel to the control unit as part of the execution of a TCW I/O operation that has the W bit set to one in the TCH. The amount of data to be transferred is specified by the DL field in the TCCB. Data is transferred in one or more transport-data IUs to the control unit on the transport exchange associated with the TCW I/O operation.

Except for the first write transport-data IU of an operation, the channel shall require a transfer-ready IU from the control unit prior to sending each transport-data IU. The amount of data sent in a transport-data IU is specified by the maximum-burst length specified in the preceding transfer-ready IU. After receiving a transfer-ready IU, the channel shall send a single transport-data IU with the amount of data specified in the preceding transfer-ready IU. The control unit shall request additional data by sending additional transfer-ready IUs until it has requested all the data specified by the DL field for the write operation. The sum of the maximum-burst length and the relative offset in a transfer-ready IU shall not exceed the value of the DL field. The maximum-burst length specified in a transfer-ready IU shall not be zero.

For the first write transport-data IU of an operation, the channel shall send the transport-data IU without receiving a transfer-ready IU if the channel is operating with first-transfer-ready disabled in effect for the operation.

First-transfer-ready disabled is in effect for an operation when both the channel and control unit indicated support for first-transfer-ready disabled during process login (see 6.3.14) and first-transfer-ready disabled has not been inhibited for the operation (see 9.3.2.2.1). If either the channel or control unit indicated they do not support first-transfer-ready disabled or if first-transfer-ready disabled has been inhibited for the operation, first-transfer-ready disabled is not in effect for the operation.

First-transfer-buffer credits is in effect for an operation when both the channel and control unit indicated support for first-transfer-buffer credits during process login (see 6.3.14 on page 47). If either the channel or control unit indicated they do not support first-transfer-buffer credits, first-transfer-buffer credits is not in effect for an operation.

If first-transfer-ready disabled is in effect for an operation, the maximum amount of data sent in the first write transport-data IU of an operation is equal to the smaller of the following:

a) if the use of first-transfer-buffer credits is not in effect, the first-burst-size specified in the PRLI accept parameters (see 6.3.14.3); or

b) if the use of first-transfer-buffer credits is in effect (see 9.3.2.2.2 on page 209), the buffer space available at the control unit as indicated by the first-transfer-buffer credit (FTBC) and the first-transfer-buffer size (FTBS); or

c) the value in the DL field.

If the channel is not operating with first-transfer-ready disabled in effect for the operation, the channel shall require a transfer-ready IU before sending the first transport-data IU.

If more than one transport-data IU is used to transfer the data, the relative offset in each transfer-ready IU shall be sequential and continuously increasing.

If the channel transfers a first transport-data IU when first-transfer-ready disabled is in effect or if the control unit requests a transport-data IU from the channel by sending a transfer-ready IU, the data sent by the channel in the transport-data IU may be for one or more write DCWs in the TCA or TCAX that have not yet become current (see 9.3.3) at the control unit.

It is model dependent whether or not a write DCW is executed by the control unit prior to receiving CRC that covers the write data for the write DCW.

For a write operation, the next IU sent by the control unit following completion of the data transfer specified by the DL field in the TCCB shall be a transport-response IU. For a bidirectional operation, the next IU sent by the control unit following completion of the data transfer specified by the DL field in the TCCB shall be a transport-data IU or a transport-response IU.

If the channel is unable to send all the data specified by the DL field due to a storage error or other condition, the channel performs the transport-mode ABTS function (see 9.4.12).

9.3.2.2.1 First Transfer Ready Disabled Inhibited

When first-transfer-ready disabled is supported by both the channel and control unit, a transport-mode operation may be performed with first-transfer-ready-disabled inhibited. When first-transfer-ready-disabled is inhibited for an operation, the channel transfers sequence initiative (see Table 9) to the control unit when it sends a transport-command IU regardless of whether first-transfer-ready disabled is supported by the channel and control unit, and only sends data after receiving a transfer-ready IU. The CU must have indicated support for either transport-mode-command retry or first-transfer-buffer credits for the channel to perform an operation with first-transfer-ready-disabled inhibited.

A transport-mode operation is performed with first-transfer-ready-disabled inhibited under any of the following conditions:

a) If the use of first-transfer-buffer credits is in effect and the number of first-transfer-buffer credits is not sufficient to send all of the data the channel intends to send in the first transport data IU (see 9.3.2.2.2); or

b) If the channel performs certain transport-command retries (see 9.5.2.2), and transport-mode command-retry is supported by both the channel and control unit; or

c) If inhibit first-transfer-ready disabled is requested by model-dependent means for a TCW I/O operation..

There may be other conditions under which a transport-mode operation is performed with first-transfer-ready-disabled inhibited.

9.3.2.2.2 First Transfer Buffer Credits

When support for first-transfer-buffer credit is indicated by both the channel and control unit during process login (see 6.3.14), the channel maintains a first-transfer-buffer credit (FTBC) and a first-transfer-buffer size (FTBS) to manage the amount of data that can be sent to the control unit in a first transport-data IU when first-transfer-ready-disabled is in effect for a transport-mode operation. The FTBC indicates the current number of first-transfer buffers that are available at the control unit and the FTBS indicates the size of each first-transfer buffer. The FTBC and FTBS applies to all transport-mode operations between a channel and control unit for all established logical paths between the channel and control unit.

Following completion of a process login in which first-transfer-buffer-credit support is indicated by both the channel and control unit, the channel sets the FTBC to the default maximum FTBC of 16 buffer credits and sets the FTBS equal to the first-transfer-buffer size specified by the control unit in the PRLI accept response.

The maximum FTBC for a channel may be modified by the control unit when it sends a transport-response IU to the channel by setting the MFTBV flag to one and by providing a different value for the maximum FTBC in the MFTBC field (see 8.15.4.11).The MFTBC provided by the control unit replaces the maximum FTBC maintained at the channel and, if the FTBC is greater than the new maximum FTBC, the FTBC is set equal to the new maximum FTBC.

When the channel sends a first transport-data IU to the control unit when first-transfer-ready disabled is in effect, it decrements the FTBC by the number of buffers required to contain all of the data sent in the transport-data IU. The channel shall not send an amount of data in a first transport-data IU that exceeds the amount of buffer space available at control unit for the channel as indicated by the FTBC maintained at the channel. When the transport-response IU is received for the operation or the operation is otherwise terminated, the channel increments the FTBC by the number of credits that were subtracted from the FTBC to perform the operation. If the resultant FTBC exceeds the maximum FTBC, the FTBC is set to the maximum FTBC.

The number of data bytes sent in the first transport-data IU is limited by the smaller of FTBC*FTBS*4K and the Data Length for the operation. The channel may send data up to this amount or may inhibit first-transfer-ready disabled for the operation. If the channel does not inhibit first-transfer-ready disabled for the operation, the amount of data sent in a first transport data IU that does not contain all of the write data for the operation shall be aligned with an integral number of First-Transfer-Buffer Credits (i.e. a multiple of FTBS). If first-transfer-ready disabled is in effect and the FTBC at the channel is zero or indicates that the first-transfer buffer space available at the control is not sufficient to contain all of the write data the channel intends to send in a first transport-data IU, the channel may inhibit first-transfer-ready disabled for the operation.

9.3.2.3 Transport Mode Read Data Transfer

A read data transfer is the transfer of data from the control unit to the channel as part of the execution of a TCW I/O operation that has the R bit set to one in the TCH. The amount of data to be transferred is specified by the DL field in the TCCB for read operations and is specified by the BRDL field for bidirectional operations. Data is transferred in one or more transport-data IUs to the channel on the transport exchange associated with the TCW I/O operation. The amount of data sent in each transport-data IU is determined by the control unit and may be set to any value as long that the total amount of data transferred in all transport-data IUs for the operation does not exceed the value in the DL field or, for bidirectional operations, the value in the BRDL field.

For read operations, if the quantity of data sent by the control unit is less than the DL field in the TCCB, the DL-residual count in the transport-response IU shall be the difference between the quantity of data sent by the CU and the DL field in the TCCB. The channel recognizes a device-level protocol error if the residual count provided by the control unit in the transport-response IU does not match the difference between the DL and the number of bytes actually received by the channel.

For bidirectional operations, if the quantity of data sent by the control unit is less than the BRDL field in the TCCB, the BRDL-residual count in the transport-response IU shall be the difference between the quantity of data sent by the CU and the BRDL field in the TCCB. The channel recognizes a device-level protocol error if the residual count provided by the control unit in the transport-response IU does not match the difference between the BRDL and the number of bytes actually received by the channel.

9.3.2.4 Transport Mode Bidirectional Data Transfer

For bidirectional operations, both write and read data transfer may be performed when bidirectional data transfer is supported by both the channel and control unit. Write data transfer is performed as described in 9.3.2.2 and read data transfer is performed as described in 9.3.2.3. Except for the first transport-data IU when first-transfer-ready disabled is in effect, the transfer of data between the channel and control unit is controlled by the control unit by either sending a read transport-data IU or by sending a transfer-ready IU to request a write transport-data IU.

The order in which data that has been received in a transport-data IU is stored to host storage in relation to the order in which data is fetched from host storage to send in a transport-data IU is unpredictable for a bidirectional operation.

When the channel transfers a first transport-data IU when first-transfer-ready disabled is in effect or when the control unit requests a transport-data IU from the channel by sending a transfer-ready IU, the data sent by the channel in the transport-data IU may be for write DCWs that are executed subsequent to the execution of read DCWs in the TCA.

9.3.3 TCA Processing

9.3.3.1 Device Command Words

The TCA consists of one or more device command words (DCW), each of which specifies an operation to be performed at the device or control unit. When the TCCB has been accepted, processing of the TCA is initiated. A TCAX may also be provided in a transport-data IU sent by the channel that contains one or more DCWs. The TCAX is considered a logical extension of the TCA.

When a TCCB is accepted by the control unit, the first DCW in the TCA becomes current and is performed by the device. The command code in the DCW indicates whether the command is a read, write, or transport command and the data count in the DCW specifies the amount of data to be transferred, if any, for the operation. All DCWs in the TCA and TCAX that transfer data shall contain a read or write command that matches the R and W bits in the TCH. For bidirectional operations, both read and write commands may be specified.

During the processing of the TCA and, if provided, the TCAX, a TCCB content error is recognized when any of the following conditions are detected:

- A DCW contains a reserved command code (see table 25).
- The first TCOB DCW or TeCOB DCW encountered is not the first DCW in the TCA;

- A second TCOB DCW or TeCOB DCW is encountered in the TCA or TCAX;
- A TCOB DCW or TeCOB DCW is encountered in the TCA when the W bit in the TCAH is zero;
- A TCOB DCW or TeCOB DCW is encountered that does not have the chain-command flag set to one;
- A TCOB DCW or TeCOB DCW is encountered in which both the CD count and data count are zero;
- A TCOB DCW or TeCOB DCW is encountered in which both the CD count and data count are non-zero;
- A TCOB DCW or TeCOB DCW is encountered in which the CD count is zero and the data count is not a multiple of 4;
- A TCOB DCW or TeCOB DCW is encountered in which the data count is zero and the CD count is not a multiple of 4;
- A TCOB DCW or TeCOB DCW is not specified and the first TTE DCW encountered is not the first DCW in the TCA or a TCOB DCW or TeCOB DCW is specified and the first TTE DCW encountered is not the second DCW in the TCA;
- A second TTE DCW is encountered in the TCA or TCAX;
- A TTE DCW is encountered in the TCA when the W bit in the TCAH is zero;
- A TTE DCW does not have the chain-command flag set to one;
- A TTE DCW is encountered in which the CD count is non-zero;
- A TTE DCW is encountered in which the data count is zero or is not a multiple of 4;
- A TTE DCW is specified and any of the following are true:
  - The TCA contains only transport-command DCWs; or
  - The last DCW in the TCA does not have the chain-command flag set to one.
- A DCW in the TCA contains a command that requires control data and the CD count field is zero or contains a value that specifies data past the end of the TCA.
- A DCW in the TCAX contains a command that requires control data and the CD count field is zero or contains a value that specifies data past the end past the end of the TCAX;
- A DCW contains a CD count field that is non-zero and the command does not allow control data to be provided;
- During a write operation, the device encountered a DCW that in the TCA or TCAX that attempted to perform a read data transfer;
- During a read operation, the device encountered a DCW in the TCA or TCAX that attempted to perform a write data transfer;

- The chain-command flag is zero in a DCW and more than 3 unused bytes remain in the TCA or TCAX;

- The chain-command flag is one in a DCW and the next DCW location is determined to be one of the following:

    - The next DCW location is within the TCA but the location is less than 8 bytes from the end of the TCA;
    - The next DCW location is beyond the length of the TCA and a TCAX has not been specified; or
    - The next DCW location is within the TCAX but the location is less than 8 bytes from the end of the TCAX.

- For a read or bidirectional operation, when a read DCW in the TCA or TCAX becomes current, the sum of the data counts for all previous read DCWs and the current DCW plus all pad and CRC bytes exceeds the DL field in the TCCB or, for bidirectional operations, exceeds the BRDL field in the TCCB;

- For a read or bidirectional operation, when the last DCW in the TCA or TCAX becomes current, the sum of the data counts for all read DCWs plus all pad and CRC bytes is not equal to the DL field in the TCCB or, for bidirectional operations, is not equal to the BRDL field in the TCCB;

- For a write or bidirectional operation, when a DCW in the TCA or TCAX becomes current, the sum of the data counts for all write DCWs plus all pad and CRC bytes (including all intermediate pad and CRC bytes), and if a COB or eCOB is included in the transport-data IU, the COB, COB pad and COB CRC bytes, or eCOB, eCOB pad and eCOB CRC bytes, and if a TCAX is present, the TCAX, TCAX pad, and TCAX CRC bytes, exceeds the DL field in the TCCB; or

- For a write operation or bidirectional operation, when the last DCW in the TCA or TCAX becomes current, the sum of the data counts for all write DCWs plus all pad and CRC bytes (including all intermediate pad and CRC bytes), and if a COB or eCOB is included in the transport-data IU, the COB, COB pad and COB CRC bytes, or eCOB, eCOB pad and eCOB CRC bytes, and if a TCAX is present, the TCAX, TCAX pad, and TCAX CRC bytes, is not equal to the DL field in the TCCB.

When the current DCW specifies command chaining and no unusual conditions have been detected during the operation, the completion of the current DCW causes the next DCW to become the current DCW and to be executed by the device. The next DCW location within the TCA is determined by adding eight plus the value in the CD-count field to the location of the current DCW and rounding to the next word boundary if not on a word boundary. When a TCAX is specified, command chaining also occurs between the last DCW in the TCA and first DCW in the TCAX, and between DCWs in the TCAX by considering the TCAX a logical extension of the TCA.

An incorrect length is detected by the control unit if the DCW data count does not match the amount of data required by the device for a write DCW or if the DCW data count does not match the amount of data available at the device for a read DCW.

If the DCW-incorrect-length facility is not supported by the channel and control unit, the processing of a TCA or TCAX is terminated and unit check status is presented if an incorrect-length condition is detected for the current DCW.

If the DCW-incorrect-length facility is supported by the channel and control unit, the processing of a TCA or TCAX for an incorrect-length condition is dependent on the state SLI flag in the DCW as described in 8.13.5.2.2.