Working Draft

X3T10.1/0989D

Information Technology -
Serial Storage Architecture - Transport Layer 1 (SSA-TL1)

Draft proposed American National Standard

This is a draft proposed American National Standard of Accredited Standards Committee X3. As such this is not a completed standard. The X3T10 Technical Committee may modify this document as a result of comments received during public review and its approval as a standard.

ABSTRACT

This standard describes the transport layer of the Serial Storage Architecture (SSA). SSA defines a serial interface hierarchy to be used for purposes within its distance and performance characteristics, including but not limited to storage subsystems.

Project Leader:
John P. Scheible
IBM Corporation.
Bldg 821 Mailstop 2835
11400 Burnet Road
Austin, TX 78758
Voice: (512) 823-8208
FAX: (512) 823-0758
EMail   scheible@vnet.ibm.com

Technical Editor:
John P. Scheible
IBM Corporation.
Bldg 902 Mailstop 9263
11400 Burnet Road
Austin, TX 78758
Voice: (512) 823-8208
FAX: (512) 838-3822
EMail   scheible@vnet.ibm.com

American National Standard
for Information Systems -

Serial Storage Architecture - Transport Layer 1 (SSA-TL1)

Secretariat
Information Technology Industry Council

Approved Month dd, yy

American National Standards Institute, Inc.

Abstract

This standard defines the transport layer of the Serial Storage Architecture (SSA). SSA defines a serial interface hierarchy to be used for purposes within its distance and performance characteristics, including but not limited to storage subsystems. This standard is intended to be used with an upper layer protocol [e. g., SCSI-2 Protocol (SSA-S2P)], and a physical layer [e. g., Physical Layer 1 (SSA-PH1)].

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer. Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made towards their resolution.

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

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

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

CAUTION: The developers of this standard have requested that holders of patents that may be required for the implementation of the standard, disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this standard.

As of the date of publication of this standard, following calls for the identification of patents that may be required for the implementation of the standard, notice of one or more claims has been received.

By publication of this standard, no position is taken with respect to the validity of this claim or of any rights in connection therewith. The known patent holder has, however, filed a statement of willingness to grant a license under these rights on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. Details may be obtained from the publisher.

No further patent search is conducted by the developer or the publisher in respect to any standard it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this standard.

Contents

Page

Annexes   Page

Figures   Page

Tables   Page

Foreword (This foreword is not part of American National Standard X3.295:1996.)

This Serial Storage Architecture - Transport Layer 1 standard is designed to provide a transport layer acceptable to vendors, looking for an evolution from parallel SCSI, and systems designers looking for opportunities to more fully exploit the capabilities inherent to a serial bus.

This standard was developed by Task Group X3T10.1 of Accredited Standards Committee X3 during 1993-96. The standards approval process started in 1995. The three informative annexes contained in this document are not part of this standard.

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent to the X3 Secretariat, Information Technology Industry Council, 1250 Eye Street, NW, Suite 200, Washington, DC 20005-3922.

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

James D. Converse, Chair

Donald C. Loughry, Vice-Chair

Joanne M. Flanagan, Secretary

Organization Represented	Name of Representative
American Nuclear Society	Geraldine C. Main
	Sally Hartzell (Alt.)
AMP, Inc	Edward Kelly
	Charles Brill (Alt.)
Apple Computer	Karen Higginbottom
Association of the Institute for Certification of Professionals (AICCP)	Kenneth Zemrowski
AT&T/NCR	Thomas W. Kern
	Thomas F. Frost (Alt.)
Boeing Company	Catherine Howells
	Andrea Vanosdoll (Alt.)
Bull HN Information Systems, Inc.	William George
Compaq Computer Corporation	James Barnes
Digital Equipment Corporation	Delbert Shoemaker
	Kevin Lewis (Alt.)
Eastman Kodak	James D. Converse
	Michael Nier (Alt.)
GUIDE International	Frank Kirshenbaum
	Harold Kuneke (Alt.)
Hewlett-Packard	Donald C. Loughry
Hitachi America, Ltd.	John Neumann
	Kei Yamashita (Alt.)
Hughes Aircraft Company	Harold L. Zebrack
IBM Corporation	Joel Urman
	Mary Anne Lawler (Alt.)
National Communication Systems	Dennis Bodson
National Institute of Standards and Technology	Robert E. Roundtree
	Michael Hogan (Alt.)
Northern Telecom, Inc	Mel Woinsky
	Subhash Patel (Alt.)
Neville & Associates	Carlton Neville
Recognition Technology Users Association	Herbert P. Schantz
	G. Edwin Hale (Alt.)
SHARE, Inc	Gary Ainsworth
	David Thewlis (Alt.)
Sony Corporation	Michael Deese
Storage Technology Corporation	Joseph S. Zajaczkowski
	Samuel D. Cheatham (Alt.)
Sun Microsystems	Scott Jameson

Technical Committee X3T10 on I/O Interfaces, which reviewed this standard, had the following members:

John B. Lohmeyer, Chair

Lawrence J. Lamers, Vice-Chair

Ralph Weber, Secretary

Mr. I. Dal Allan	Mr. Paul D. Aloisi	Mr. David Andreatta
Mr. Tak Asami	Mr. Akram Atallah	Mr. Wayne Baldwin
Mr. Geoffrey Barton	Mr. Robert Bellino	Mr. Rick Bohn
Mr. Paul Boulay	Mr. Charles Brill	Mr. Michael Bryan
Mr. John Cannon	Mr. Ting Li Chan	Mr. Shufan Chan
Mr. Joe Chen	Mr. Andy Chen	Mr. Jack Chen
Ms. Nancy Cheng	Mr. Mike Chennery	Mr. Dan Colegrove
Mr. Roger Cummings	Mr. Chris D'Iorio	Mr. Zane Daggett
Mr. William Dallas	Mr. Joe Dambach	Mr. Brian N. Davis
Mr. Jan V. Dedek	Mr. Dhiru N. Desai	Mr. Mike Eneboe
Mr. Mark Evans	Mr. Timothy Feldman	Mr. Stephen G. Finch
Mr. Edward Fong	Mr. Edward A. Gardner	Mr. John Geldman
Mr. Raymond Gilson	Mr. Chuck Grant	Mr. Louis Grantham
Mr. Dave Guss	Mr. Peter Haas	Mr. Douglas Hagerman
Mr. Kenneth J. Hallam	Dr. William Ham	Mr. Tom Hanan
Mr. Norm Harris	Mr. Edward Haske	Mr. Stephen F. Heil
Mr. Mike Hetzel	Mr. Stephen Holmstead	Mr. Gerald Houlder
Mr. Peter Johansson	Mr. Gerry Johnsen	Mr. Brian Johnson
Mr. Skip Jones	Mr. Richard Kalish	Mr. Greg Kapraun
Mr. Thomas J. Kulesza	Mr. Lawrence J. Lamers	Mr. Dennis Lang
Mr. Edward Lappin	Mr. Pat LaVarre	Mr. Jaff Lin
Mr. Robert Liu	Mr. John Lohmeyer	Mr. Bill Mable
Mr. John Masiewicz	Mr. David McFadden	Mr. James McGrath
Mr. Pete McLean	Mr. Patrick Mercer	Mr. Gene Milligan
Dr. Akira James Miura	Mr. E.J. Mondor	Mr. Charles Monia
Mr. Ian Morrell	Mr. John Moy	Mr. S. Nadershahi
Mr. Jay Neer	Mr. Marc A. Noblitt	Mr. Tim Norman
Mr. Vit Novak	Mr. Erich Oetting	Mr. Dennis Pak
Mr. Duncan Penman	Mr. George Penokie	Mr. Doug Piper
Mr. Kevin R. Pokorney	Mr. Gary Porter	Mr. Robert Reisch
Mr. Ron Roberts	Mr. Frank Samela	Mr. John P. Scheible
Mr. Steve D. Schueler	Mr. J. R. Sims	Mr. Michael Smith
Mr. Robert N. Snively	Mr. Jeff Stai	Mr. Gary R. Stephens
Mr. Arlan P. Stone	Mr. Clifford E. Strang Jr.	Mr. Nicos Syrimis
Mr. Matthew Thomas	Mr. Pete Tobias	Mr. Tokuyuki Totani
Mr. Dennis Van Dalsen	Mr. Joseph Wach	Mr. Dean Wallace
Mr. Roger Wang	Mr. Gary M. Watson	Mr. Ralph O. Weber
Mr. Bob Whiteman	Mr. Jeffrey L. Williams	Mr. Michael Wingard
Mr. Devon Worrell	Mr. Anthony Yang	Mr. Danny Yeung
Mr. Ruben Yomtoubian		

Task Group X3T10.1 on Serial Storage Architecture, which developed this standard, had the following members:

Lawrence J. Lamers, Chair

John Scheible, Vice-Chair

Bob Atkinson	James M. Avery	Robert Bellino
Paul Boulay	Charles Brill	Phil Bryan
Ed Carmona	Nancy Cheng	David Deming
Dhiru N. Desai	Mark DeWilde	Neil Edmunds
Richard Egan	Ken Erickson	Greg Farrin
Chuck Gibson	Chuck Grant	Ken Hallam
Bill Ham	Adge Hawes	Greg Kapraun
Sam Karunanithi	Brad Kitson	Michael Knowles
Lawrence Lamers	John Lohmeyer	Bill Mable
Gary Manchester	Charles Monia	Rich Moore
Ian Morrell	Karl Nakamura	Jay Neer
Thai Nguyen	Eric Oetting	Paul Petersen
Kevin Pokorney	Said Rahmani	Monica Roy
Sam Sanyal	John Scheible	Pete Tobias
Douglas Wagner	Michael Wingard	Kevin Ziegler

Introduction

This standard is divided into the following clauses and annexes.

Clause 1 defines the scope of the Serial Storage Architecture - TL1 (transport layer).

Clause 2 specifies the normative references.

Clause 3 defines the definitions, symbols and abbreviations.

Clause 4 contains the conventions.

Clause 5 defines the Web.

Clause 6 defines the characters.

Clause 7 defines the frames.

Clause 8 defines the protocol.

Clause 9 defines the link management.

Clause 10 defines the error handling.

Clause 11 defines the SMSs.

Clause 12 defines the processes.

Annex A is informative and defines the data transfer processes.

Annex B is informative and defines the services.

Annex C is informative and shows examples of half and full duplex operation

Draft American National Standard
for Information Systems -

Information Technology -
Serial Storage Architecture
Transport Layer 1 (SSA-TL1)

1   Scope

This document defines the transport layer of the Serial Storage Architecture (SSA). SSA defines a serial
interface hierarchy to be used for purposes within its distance and performance characteristics, including but
not limited to storage subsystems. This standard is intended to be used with an upper layer protocol [e. g.,
SCSI-2 Protocol (SSA-S2P)] and a physical layer [e. g., SSA Physical Layer 1 (SSA-PH1)].

A major goal of the SSA-TL1 standard is to define a transport layer acceptable to vendors looking for an
evolution from parallel SCSI and systems designers looking for opportunities to more fully exploit the
capabilities inherent to a serial bus.

1.1   SSA-TL1 characteristics

The essential characteristics of SSA-TL1 are as follows:

   a) SSA-TL1 is optimized to minimize gate count while maintaining function and performance.
   b) SSA-TL1 provides a flexible addressing scheme to allow the configuration of a Web into dedicated
       connections, strings, loops, and complex configurations including a switch.
   c) The Web supports peer-to-peer communication with frame multiplexing between any pair of ports.
   d) The Web is self-configuring without the use of address switches.
   e) The link supports point-to-point physical connections only. Strings and loops are created by using
       dual port nodes with cut-through routing.
   f) The transport layer flow control allows operation from 5 MB/s at a distance of 2,5 km to 13 600 MB/s at
       a distance of 1 m. The actual data rates and distances depend on the cable assembly and the drivers
       and receivers used.
   g) The link makes optimum use of the physical medium by using full-duplex communication to avoid
       arbitration overhead and turn-around delay.
   h) Full duplex communication achieves an aggregate 40 MB/sec bandwidth (20 MB/sec in each direction)
       between two ports.
   i) The frames are constructed with a minimum framing overhead (e.g., only 8 characters of overhead for
       a 128 byte data field).
   j) An architected error recovery procedure provides transparent frame recovery after a transmission
       error.
   k) A Wrap mode is provided to allow fault isolation at a node.
   l) An optional line fault detector detects failures in the link.
   m) Cyclical paths allow a Web to be configured providing redundant access to a node.
   n) The transport layer is capable of simultaneously supporting multiple upper-level protocols (e.g., SCSI-
       2, SCSI-3, IPI-3, TCP/IP, etc.).
   o) The 128-byte frame size minimizes the gate count needed for buffers.

This standard defines the following functions:

    a)   the protocol (e.g., framing, addressing and flow control);

    b)   link management (e.g., buffering, port states, resets, configuration and error recovery).

1.2 SSA family of standards

The relationships of the SSA family of standards is illustrated in Figure 1.

```
┌─────────────────────────┐        ┌─────────────────────────┐
│         SSA-S2P         │        │         SSA-S3P         │
│ Serial Storage Architecture│     │ Serial Storage Architecture│
│      SCSI-2 Protocol    │        │      SCSI-3 Protocol    │
└─────────────────────────┘        └─────────────────────────┘

┌─────────────────────────┐        ┌─────────────────────────┐
│         SSA-TL1         │        │         SSA-TL2         │
│ Serial Storage Architecture│     │ Serial Storage Architecture│
│      Transport Layer 1  │        │      Transport Layer 2  │
└─────────────────────────┘        └─────────────────────────┘

┌─────────────────────────┐        ┌─────────────────────────┐
│         SSA-PH1         │        │         SSA-PH2         │
│ Serial Storage Architecture│     │ Serial Storage Architecture│
│      Physical Layer 1   │        │      Physical Layer 2   │
└─────────────────────────┘        └─────────────────────────┘
```
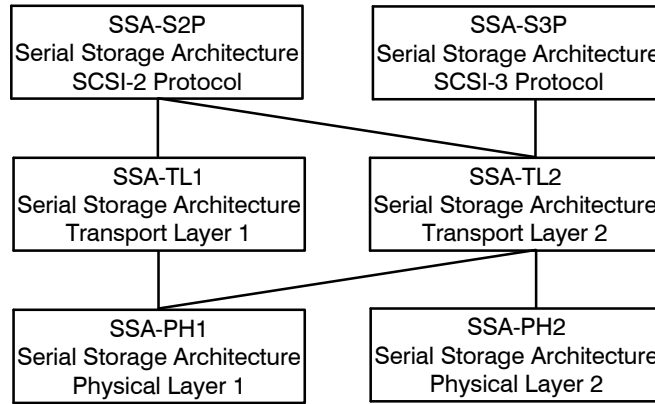
Figure 1 - Relationship of the SSA standards

1.2.1 SSA-S2P

Serial Storage Architecture - (SSA-S2P) SCSI-2 Protocol defines the SCSI-2 Protocol used with SSA-TL1 or SSA-TL2.

1.2.2 SSA-S3P

Serial Storage Architecture - (SSA-S3P) SCSI-3 Protocol defines the SCSI-3 Protocol used with SSA-TL2.

1.2.3 SSA-TL1

Serial Storage Architecture - (SSA-TL1) Transport Layer 1 defines the Transport layer that supports SSA-S2P and requires SSA-PH1.

1.2.4 SSA-TL2

Serial Storage Architecture - (SSA-TL2) Transport Layer 2 defines the Transport layer that supports SSA-S2P and SSA-S3P and requires SSA-PH1 or SSA-PH2.

1.2.5 SSA-PH1

Serial Storage Architecture - (SSA-PH1) Physical Layer 1 defines the Physical layer that supports SSA-TL1 and SSA-TL2, and consists of the electrical characteristics of the interface and the connectors.

1.2.6 SSA-PH2

Serial Storage Architecture - (SSA-PH2) Physical Layer 2 defines the Physical layer that supports SSA-TL2, and consists of the electrical characteristics of the interface and the connectors.

2 Normative references

The following standard contains provisions which, through reference in SSA-S2P, constitute provisions of this standard. At the time of publication, the edition indicated was 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 standard listed below. Members of IEC and ISO maintain registers of currently valid International Standards, and ANSI maintains registers for American National Standards.

ANSI X3.294:1996, Information Technology, Serial Storage Architecture - SCSI-2 Protocol (SSA-S2P)

ANSI X3.293:1996, Information Technology, Serial Storage Architecture - Physical Layer 1 (SSA-PH1)

3 Definitions, symbols and abbreviations

3.1 Definitions

3.1.1   application: A process that is communicating via the link.

3.1.2   APPLICATION: A possible value of the FRAME TYPE field.

3.1.3   Application frame: A frame with a FRAME TYPE field value of APPLICATION.

3.1.4   channel: The facilities in a port to receive an SMS or a single data transfer.

3.1.5   character: A sequence of 10 encoded bits that represents a data byte or a protocol function.

3.1.6   Configutor node: A node that knows the topology of the Web by performing the Configuration process. If a node is not a Configutor node, then it is a Responder node.

3.1.7   CONTROL: A possible value of the FRAME TYPE field.

3.1.8   CONTROL field: The first byte of a frame.

3.1.9   Control frame: A frame with a FRAME TYPE field value of CONTROL.

3.1.10   cut-through frames: Frames that are not addressed to the receiving node and are passed from the input lines of one port to the output lines of another port with minimal latency according to the routing rules.

3.1.11   cut-through routing: A process where a frame not addressed to a node is routed to the output lines of another port as it streams in the input lines of the receiving port.

3.1.12   cyclic path: A portion of a Web where starting at a given node, it is possible to return to that same node after traversing one or more links without traversing any node more than once. A loop is an example of a cyclic path.

3.1.13   Data frame: An application frame with a non-zero Channel component.

3.1.14   destination node: The node where the frame arrives with an path component of 00h.

3.1.15   Dual port node: A node with two ports and a cut through router between them.

3.1.16   field: A group of related data characters in a frame (e.g., the CRC field).

3.1.17   forward: The act of receiving a frame not addressed to the node on one port and transmitting out another port after modifying the frame according to the protocol rules.

3.1.18   frame: A sequence of 6 or more data characters surrounded by FLAG characters.

3.1.19   line: A physical connection between a transmitter and a receiver.

3.1.20   link: A serial connection between two ports.

3.1.21   loop: A cyclic Web containing dual port nodes only.

3.1.22   Master: A unique node that coordinates error recovery.

3.1.23 meaning code: The description assigned the combination of the TYPE, SUBTYPE, and TYPE INFO fields used as a Link ERP exit, in the ASYNC ALERT SMS or MASTER ALERT SMS.

3.1.24 node: A system, controller, switch or device with one or more ports.

3.1.25 Originating frames: Frames that are generated from the node itself and are sent out the output lines on one of the ports of the node.

3.1.26 path: The links and intermediate nodes used to transfer a frame from the source to the destination.

3.1.27 port: The hardware and firmware that implements one end of a link.

3.1.28 PRIVILEGED: A possible value of the FRAME TYPE field.

3.1.29 Privileged frame: A frame with a frame type field value of PRIVILEGED.

3.1.30 receiver, Rx: The logic that decodes the signal on the inbound line.

3.1.31 remote port: The port connected to the other end of the SSA link.

3.1.32 RESERVED: A possible value of the FRAME TYPE field or the RESET TYPE field.

3.1.33 reserved field: A field that is reserved for future standardization.

3.1.34 reserved field value: A field value that is reserved for future standardization.

3.1.35 Reserved frame: A frame with a frame type field value of RESERVED.

3.1.36 Reserved Control frame: A frame with a frame type field value of CONTROL, and a RESET TYPE field value of RESERVED.

3.1.37 Responder node: A node with no knowledge of the configuration. A Responder node shall be given a RETURN PATH or RETURN PATH ID to send a response, since it does not have knowledge of the Web topology. If a node is not a Responder node, then it is a Configutor node.

3.1.38 source node: The node that originates a particular frame.

3.1.39 SMS: The DATA field portion of a frame with a channel field of 00h.

3.1.40 spatial reuse: The property where multiple sets of transfers between source and destination pairs are allowed on the Web at any given time.

3.1.41 string: A linear Web of dual port nodes. (The nodes at either end shall be a switch, a single port node or a dual port node with one port not operational.)

3.1.42 transmitter, Tx: The logic that drives the outbound line.

3.1.43 Web: A collection of SSA nodes addressable by each other that are connected by links (i.e., a loop, a string, or complex configurations).

3.2 Symbols and abbreviations

CMOS   complementary metal oxide semiconductor.
CRC   cyclic redundancy check.
DMA   direct memory access.
ERP   error recovery procedure.
FCS   fiber channel standard.
FDDI   fiber distributed data interface.
FSN   frame sequence number.
LED   light emitting diode.
LSI   large scale integration.
POR   power-on reset.
POST   power-on self-test.
RAS   reliability, availability and serviceability.
RSN   receive sequence number.
SCSI   Small Computer Systems Interface.
SSA   Serial Storage Architecture.

4

SMS   SSA Message Structure.
TSN   transmit sequence number.
&   logical AND
=   assignment or comparison for EQUAL
≠   comparison for NOT EQUAL
<   comparison for LESS THAN
≤   comparison for LESS THAN OR EQUAL TO
>   comparison for GREATER THAN
+   ADD
-    SUBTRACT
*   MULTIPLY
±   PLUS OR MINUS
≈   APPROXIMATELY
»   MUCH GREATER THAN

4 Conventions

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 glossary or in the text where they first appear. Lower case is used for words having the normal English meaning.

Fields containing only one bit are usually referred to as the "named" bit instead of the "named" field. When a bit is set its value is 1. When a bit is cleared, its value is 0.

Numbers that are not immediately followed by lower-case "b" or "h" are decimal values.

Numbers immediately followed by lower-case "b" (xxb) are binary values.

Numbers immediately followed by lower-case "h" (xxh) are hexadecimal values.

Decimal numbers are indicated with a comma(e.g., two and one half is represented as "2,5"). Decimal numbers having a value exceeding 999 are represented with a space (e.g., 24 255).

The bit ordering within a byte used in SSA-TL1 is illustrated in Table 1.

Table 1 - Bit ordering in a byte

Bit 7	6	5	4	3	2	1	Bit 0
msb							lsb

The bits in an encoded character are designated a b c d e i f g h j. Bit a is transmitted on the line first and the other bits follow in the order shown.

Reserved bits, fields, bytes, and code values are set aside for future standardization. Their use and interpretation may be specified by future extensions to this standard. A reserved bit, field, or byte shall be set to zero, or in accordance with a future extension of this standard. Reserved bits, fields, bytes or reserved field values shall be ignored when cut-through routing a frame. A destination node that receives a reserved field value in the FRAME TYPE or RESET TYPE fields of the CONTROL field of a frame shall generate a FRAME REJECT ERROR as defined in 10.1.1.9. A destination node that received an SMS with a reserved bit, field, or byte that is not zero, or receives a reserved code value shall respond as defined in 10.2.7.

Ignored bits, fields, or bytes shall be ignored by the receiving node. Any value shall be considered valid.

SMS names are shown as all capital letters, such as QUERY NODE SMS. Field names are shown as small capital letters, such as the FRAME TYPE field. Field values are shown as all capital letters, such as the FRAME TYPE field APPLICATION FRAME value. Variable names are in italics, such as N.

The byte ordering convention is Big Endian (i.e. the most significant byte of a number is sent first).

SSA-PH is used to refer to either SSA-PH1 or SSA-PH2. SSA-TL is used to refer to either SSA-TL1 or SSA-TL2.

5 Web

An SSA Web consists of two or more nodes interconnected by links. Each node has one or more ports. A port consists of the hardware and firmware to support one end of a link.

A node is identified in several ways.

   a) Each node is either a Configutor or a Responder, and this determines the processes and SMS codes that shall be supported. A Configutor node knows the topology of the Web but a Responder node does not.

   b) At any time, at most one node in the Web is a Master, who is responsible for coordinating error recovery. Every Configutor node shall be capable of being a Master node, but a Responder node shall not.

   c) A node has one port ("Single port"), two ports ("Dual port"), or more than two ports ("Switch").

Dual port nodes and switches contain a router between the ports and the node function. Figure 2 shows a dual port node.
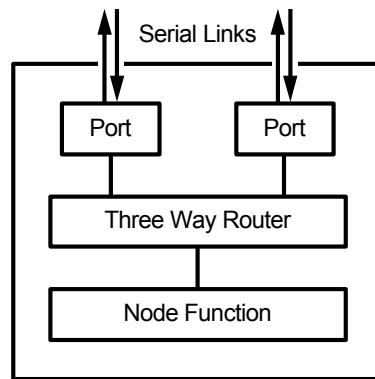
Figure 2 - Dual-port node

Depending on the ADDRESS field, the router forwards an inbound frame to the node function or to the outbound line of the other port. When the dual port node originates a frame it instructs the router to transmit it via the specified port.

The following clauses describe the types of Webs possible.

5.1 Strings

A string is a linear Web of two or more nodes, as shown in Figure 3. The node at either end of a string shall be a single port node, a dual port node with one port not operational or a switch. The intermediate nodes (if any) shall be dual port nodes.
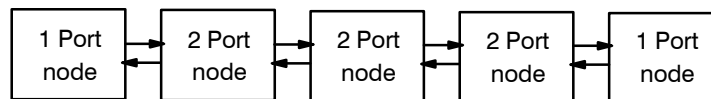
Figure 3 - A string

Each link operates independently, managing its own flow control and error recovery. An error on one link does not affect the other links, except to abort a frame that is already being forwarded.

To allow unrestricted communication between any two nodes the maximum number of nodes in a string is 129, including the end nodes.

The simplest case of a string is a dedicated connection between 2 single port nodes, as shown in Figure 4.
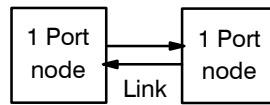
Figure 4 - A dedicated connection

5.2 Loops

A loop is a cyclic Web containing only dual port nodes, as shown in Figure 5. A loop shall not contain a switch because a switch breaks the continuity of the loop.
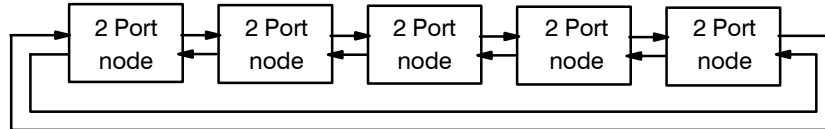
Figure 5 - A loop

A loop provides better availability than a string because the failure of a dual port node or link does not block the communication between any pair of the remaining nodes. The dynamic insertion or removal of a dual port node does not prevent communication between the other nodes.

To allow the Configuration process to determine that a loop is closed, the maximum number of nodes is limited to 128.

An important property of SSA is spatial reuse. Each link functions independently of the other links so that several transfers occur concurrently at the full bandwidth provided that each uses different links. This allows a much higher through-put than a typical bus.

Fairness algorithms are used to share the link bandwidth equitably when there is contention for the use of a particular link (see 8.5).

5.3 Complex web containing switches

Switches allow strings to be connected to allow over 65 000 ports to intercommunicate with full error reporting capability. Switches also allow alternative paths to be provided to achieve fault tolerance. A complex Web including 2 switches and 5 strings, where one string is a cyclic path, is shown in Figure 6.
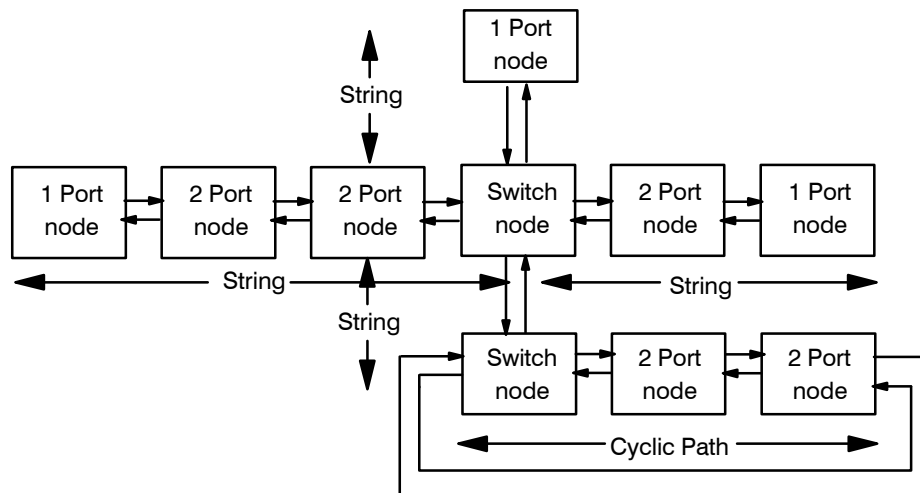
Figure 6 - A complex Web containing a cyclic path

The maximum number of addressable ports in a switch is 126. However this maximum is reduced when the path preceding the switch is a string. For example, if a string of 40 nodes leads to a switch then the maximum number of switch ports addressable by the first node is 86.

6 8B/10B code

Data bytes and protocol functions are encoded into 10-bit characters for transmission on the physical medium.

Synchronous clocking restricts the bit patterns that the transmitter uses since it is undesirable to have long sequences of consecutive zeros or ones. Therefore an encoding algorithm is required to convert the arbitrary data that one wishes to send into patterns suitable for transmission.

The link uses an 8B/10B code[1]. The 8B/10B code provides a (0,4) run-length constraint. This means that the minimum run-length of consecutive zeros or ones is 1 bit and the maximum run-length is 5 bits. The 8B/10B code also provides DC-balance with a maximum digital sum variation of 6.

The 8B/10B mapping is achieved efficiently by partitioning each incoming byte into a 5-bit sub-block and a 3-bit sub-block. The sub-blocks are then encoded with separate 5B/6B and 3B/4B encoders.

In order for the encoded data to be DC-balanced, the combined 8B/10B encoder accumulates a running disparity. As a result, the encoder and decoder require sequential logic in addition to combinatorial logic. The disparity of a sub-block is the difference between the number of 1s and 0s in the encoded data; positive and negative disparities refer to an excess of 1s and 0s respectively. For both the 6B and the 4B sub-blocks the permitted disparity is 0, +2 or -2. The 8B/10B code requires that the polarity (positive or negative) of encoded sub-blocks with non-zero disparity shall alternate. The 4B and 6B encodes with non-zero disparity are allocated in complementary pairs to allow this. With a few exceptions, encoded sub-blocks with zero disparity are generally independent of the running disparity.

6.1 Data characters

Bits 7:3 are encoded into bits a, b, c, d, e, i by the 5B/6B encoder using the rules in Table 2.

The first encode is used when the polarity of the running disparity on entry to the sub-block is as shown in the column D-1. The polarity of the running disparity is equal to the polarity of the most-recent sub-block that had non-zero disparity. An x in column D-1 indicates that the first encode is selected independently of the entry disparity. The column D0 indicates the disparity of the first encode, whose value shall be 0, +2 or -2.

The alternate encode is used if the polarity of the running disparity does not match the entry in column D-1. In this case the encode disparity is the complement of column D0.

---

[1]For informative reference only. A. X. Widmer and P. A/ Franazek, "A DC-Balanced, Partition-Block, 8B/10B Transmission Code", IBM Journal of Research and Development, Vol. 27 no. 5, September 1983, pp. 440-450.

Table 2 - 5B/6B encoding rules

Symbol	Input 76543	D-1	Encode abcdei	D0	Alternate abcdei
D.0	00000	+	011000	-	100111
D.1	10000	+	100010	-	011101
D.2	01000	+	010010	-	101101
D.3	11000	x	110001	0	
D.4	00100	+	001010	-	110101
D.5	10100	x	101001	0	
D.6	01100	x	011001	0	
D.7	11100	-	111000	0	000111
D.8	00010	+	000110	-	111001
D.9	10010	x	100101	0	
D.10	01010	x	010101	0	
D.11	11010	x	110100	0	
D.12	00110	x	001101	0	
D.13	10110	x	101100	0	
D.14	01110	x	011100	0	
D.15	11110	+	101000	-	010111
D.16	00001	-	011011	+	100100
D.17	10001	x	100011	0	
D.18	01001	x	010011	0	
D.19	11001	x	110010	0	
D.20	00101	x	001011	0	
D.21	10101	x	101010	0	
D.22	01101	x	011010	0	
D.23	11101	-	111010	+	000101
D.24	00011	+	001100	-	110011
D.25	10011	x	100110	0	
D.26	01011	x	010110	0	
D.27	11011	-	110110	+	001001
D.28	00111	x	001110	0	
D.29	10111	-	101110	+	010001
D.30	01111	-	011110	+	100001
D.31	11111	-	101011	+	010100

Bits 2:0 are encoded into bits f, g, h, j by the 3B/4B encoder using the rules in Table 3.

Table 3 - 3B/4B encoding rules

Symbol	Input 210	D-1	Encode fghj	D0	Alternate fghj
D.0	000	+	0100	-	1011
D.1	100	x	1001	0	
D.2	010	x	0101	0	
D.3	110	-	1100	0	0011
D.4	001	+	0010	-	1101
D.5	101	x	1010	0	
D.6	011	x	0110	0	
D.P7	111	-	1110	+	0001
D.A7	111	-	0111	+	1000

This is interpreted in the same way as the 5B/6B table.

Note that although symbols D.7 in the 5B/6B encodes and D.3 in the 3B/4B encodes have zero disparity they are assigned complementary encodes. This reduces the maximum digital sum variation and eliminates certain undesirable long run lengths.

The encode for bits 2:0 = 111b in the 3B/4B encoder has two possible values. Symbol D.A7 shall replace D.P7 under either of the following conditions:

    a)   in the previous sub-block e = i = 1b and the running disparity is negative;

    b)   in the previous sub-block e = i = 0b and the running disparity is positive.

6.2 Special characters

The 8B/10B code provides 12 Special characters as defined in Table 4.

Table 4 - Special characters

Symbol	Function	D-1	Encode abcdei fghj	D0	Alternate abcdei fghj
K.28.0[1]	User defined	-	001111 0100	0	110000 1011
K.28.1[2]	FLAG	-	001111 1001	+	110000 0110
K.28.2	ABORT	-	001111 0101	+	110000 1010
K.28.3	SAT	-	001111 0011	+	110000 1100
K.28.4	SAT'	-	001111 0010	0	110000 1101
K.28.5[2]	DIS	-	001111 1010	+	110000 0101
K.28.6[1]	User defined	-	001111 0110	+	110000 1001
K.28.7[2]	Invalid	-	001111 1000	0	110000 0111
K.23.7	ACK	-	111010 1000	0	000101 0111
K.27.7	RR	-	110110 1000	0	001001 0111
K.29.7	NUL	-	101110 1000	0	010001 0111
K.30.7[1]	User defined	-	011110 1000	0	100001 0111

NOTES -
[1] - These symbols are defined in 6.2.9.
[2] - The symbol column denotes a comma.

Three of the special characters contain a sequence (bits a, b, c, d, e, i, f = 0011111b) that does not occur in any bit position, either within other characters or in any overlap of characters. They are called Comma characters and their properties make them suitable for establishing byte synchronization.

SSA-TL1 uses 8 of the Special characters as Protocol characters. The special character K28.7 is defined as an Invalid character since K28.7 leads to the detection of a false comma, and shall be handled as a code violation (see 6.4). The other 3 special characters are User Defined characters that are available for use by upper-level protocols.

6.2.1 DIS character

The DIS character is sent when a port is in the Disabled state and is used to establish character synchronization (see 9.1.3.1).

6.2.2 FLAG character

In the Enabled state, the receipt of a FLAG character places the port into the Ready state (see 9.1.3.2). Provided that character synchronization has been achieved the following actions are taken when a FLAG character is detected.

    a)   If a data character has been received since the previous FLAG character then the receiver considers the current frame to be complete.

    b)   The receiver becomes ready to receive a frame.

FLAG characters are also sent when the link is idle in order to maintain character synchronization by the receiver.

6.2.3 NUL character

The transmitter may insert NUL characters within a frame anywhere after the CONTROL field. A NUL character shall not be sent between the leading FLAG character and the CONTROL field of a frame. Also a NUL character

shall not be sent between frames or in the Disabled state. For example, NUL characters may be inserted within the ADDRESS, DATA and CRC fields of a frame, and within Control frames.

Except for protocol errors, the receiver shall ignore NUL characters by discarding them without changing its state. NUL characters shall not be included when calculating the frame CRC. A node shall insert NUL characters in a cut-through frame when data for that frame is not available from the input port (e.g. NUL characters received).

NUL characters are used in the following cases.

a) If the transmitter has started to send a frame but the data needed to complete the frame is temporarily unavailable.

b) If the transmitter is still waiting for an ACK character pair when it is ready to send the trailing FLAG character of the next frame (see 8.4.1).

6.2.4 RR character

The RR characters (Receiver Ready characters) shall always be sent in pairs. An RR character pair is used to pace the transmission of frames, and allows the remote port to send another frame (see 8.4.3).

6.2.5 ACK character

The ACK characters (Acknowledge characters) shall always be sent in pairs. An ACK character pair is used to acknowledge the receipt of a valid frame (see 8.4.1).

6.2.6 ABORT character

A port may abort a frame after it has started to transmit the frame by sending a single ABORT character followed immediately by the trailing FLAG character. The remote port discards the frame as though it had never been sent.

> NOTE 1 - If a link error corrupts an ABORT character into a NUL or a FLAG character following an otherwise valid frame, then the receiver accepts the frame as valid. To protect against this it is recommended that the transmitter ensure the accumulated CRC is bad before sending the ABORT character. If the CRC has already been sent then it is recommended that the transmitter send an extra data character before the ABORT character. This extra data character causes the receiving node to use the last three bytes of CRC and the extra byte as the CRC field thereby almost guaranteeing bad CRC is detected. This gives three levels of indicating a bad frame (ABORT characters, bad CRC, and extra byte). However, some hardware implementations leave a small window when bad CRC cannot be guaranteed, so this is an implementer's note.

If the ABORT character is received when the frame is already being forwarded, then the ABORT character shall also be forwarded.

6.2.7 SAT character

The SAT character (Satisfied character) is used to balance between spatial reuse and fairness of the Web using the SAT algorithm. The receipt of a SAT character authorizes a port that has not transmitted its quota of frames to hold Cut-through frames and give frames it originates a higher priority as specified by the SAT algorithm (see 8.5).

6.2.8 SAT' character

The SAT' character is used in string configurations and is immediately forwarded or reflected by the port as part of the SAT algorithm. The SAT' character does not influence frame origination (see 8.5).

6.2.9 User defined characters

SSA-TL1 provides three User Defined characters for use by upper-level protocols. However, to avoid conflicts when more than one upper-level protocol is operating on the same Web, the definition of these characters shall be documented in SSA-TL1 transport layer.

The SYNC character is used as an Index pulse to synchronize multiple devices.

The forwarding of User Defined characters shall be selectively enabled for each port (see 11.2.3).

A User Defined character is processed if the input port is in the Ready or Check states. A User Defined character may be forwarded when the output port is in the Ready or Check states.

A User Defined characters are defined in Table 5.

Table 5 - User Defined character definition

Number	K Code	Definition	Used by
1	K28.0	SYNC	SSA-S2P
2	K28.6	reserved	reserved
3	K30.7	reserved	reserved

6.3 Initialization

Until character synchronization has been achieved the running disparity of the transmitter is not known to the receiver. Therefore when the receiver first acquires character synchronization in the Disabled state it shall accept one FLAG or DIS character with either disparity. This character establishes the running disparity in the receiver. To be considered valid subsequent characters shall then follow the disparity rules.

The receiver shall follow a similar procedure to re-initialize its running disparity after it has detected a disparity error.

6.4 Code violations

The receiver shall check for illegal 6B sub-blocks, 4B sub-blocks or the K28.7 characters that are outside of the defined data or Special characters. The receiver shall also check that each sub-block and special character is consistent with the required entry disparity. The error leading to a violation may have occurred in a previous character. Finally, the receiver shall check that the 4B sub-blocks D.P7 and D.A7 are consistent with the previous 6B sub-block.

Code violations are only checked when the port is in the Ready state, and are ignored in all other states. If the receiver detects an error while the node is in the Ready state it shall generate a Link Reset frame with a RECEIVER ERROR field value of CODE VIOLATION in the Link Status Byte (see 10.1.1.5).

6.5 Permitted special character sequences

This clause defines the restrictions on the transmission of Special characters.

The following pairs of protocol characters are indivisible (i.e., the first character is valid only when followed immediately by the second character):

   a)   ACK - ACK;
   b)   RR - RR;
   c)   ABORT - FLAG.

There shall be at least one data character between an ABORT - FLAG pair and the last preceding FLAG character (e.g., FLAG, ABORT, FLAG is not allowed).

Violation of these rules while in the Ready state shall cause the receiver to invoke the Link ERP process and exit with a meaning code of PROTOCOL ERROR (see 10.1.1.6).

The following protocol characters may be inserted anywhere within a frame. They may also be inserted within an idle sequence of FLAG characters except during the first 10 FLAG characters when a port is in the Ready or Check states:

   a)   ACK - ACK;
   b)   RR - RR;
   c)   User Defined characters;
   d)   SAT and SAT'.

7 Frames

The two ports at opposite ends of a link communicate in units called frames. A frame consists of a sequence of at least 6 data characters delimited by FLAG characters at each end. The FLAG characters that mark the beginning and end of a frame are referred to as the leading and trailing FLAGS respectively. The first data character following a FLAG character marks the beginning of a frame. The last data character preceding a FLAG character marks the end of a frame. A frame is divided into a sequence of 3 or 4 fields as shown in Figure 7.

To minimize overheads the trailing FLAG character may act as the leading FLAG character of the next frame. Therefore to ensure compatibility all nodes shall be able to receive consecutive frames that are separated by a single FLAG.
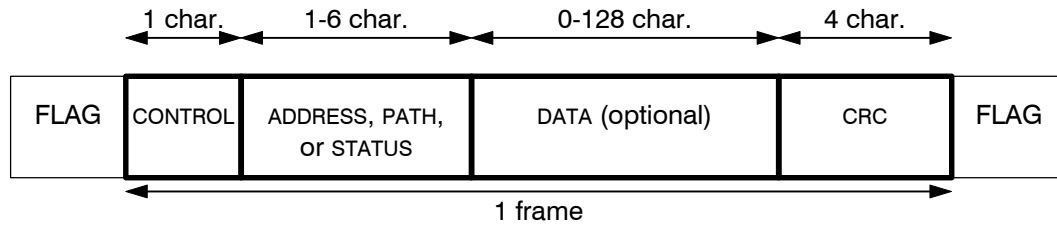
Figure 7 - Generalized frame format

The maximum lengths of the ADDRESS and DATA fields were chosen to balance between Web size, communication efficiency and implementation cost.

A valid frame is a frame that was not aborted by an Abort character, and does not invoke Link ERP at the receiving port (see 10.1).

The transport layer supports three types of frames: Control, Privileged, and Application frames. The frame type is controlled by the FRAME TYPE field (see 7.1.1).

A Control frame is used for resets. If the length of the DATA field of a Control frame is not zero, the receiving node rejects the frame. A Control frame shall be acted upon immediately by the destination node and is not subject to the pacing rules (see 8.4.3). Control frames are only sent when the transmitting port is in the Ready, Enabled or Check states and in Privileged or Normal mode (see 9.1.3 and 9.1.4).

A Privileged frame is used by the transport layer for configuration and error recovery. Privileged frames shall be subject to the pacing rules, and are only sent when the transmitting port is in the Ready state and in Privileged or Normal mode.

An Application frame is used by the transport layer for data transfer and to transport any SMSs sent by the upper-layer protocol. Application frames shall be subject to the pacing rules, and are only sent when the transmitting port is in the Ready state and Normal mode. When a port is in Privileged mode the transmitter and receiver discard Application frames.

The content of a frame depends on the control field as shown in Table 6 and Figure 8.

Table 6 - Frame contents

CONTROL field		ADDRESS, PATH, STATUS field	DATA field	CRC field
FRAME TYPE	FSN or RESET TYPE			
APPLICATION	FSN	Path and Channel	Data	CRC
RESERVED				CRC
PRIVILEGED	FSN	Path and Channel	Data	CRC
CONTROL	LINK RESET	Link Status Byte	not present	CRC
CONTROL	TOTAL RESET	Path	not present	CRC
CONTROL	RESERVED			CRC
CONTROL	ABSOLUTE RESET	Path	not present	CRC

Privileged, Reserved, or Application frame

| CONTROL | ADDRESS | DATA | CRC |

Total Reset,Reserved, or Absolute Reset Control frame

| CONTROL | PATH | CRC |

Link Reset frame

| CONTROL | STATUS | CRC |
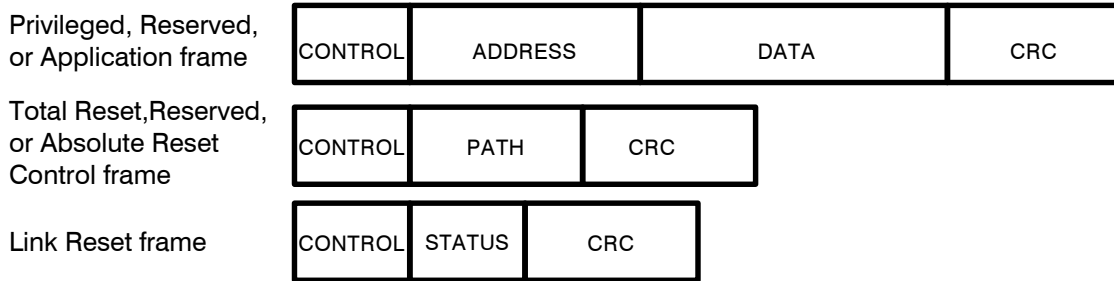
Figure 8 - Specific frame format

7.1 CONTROL field

The CONTROL field defined in Table 7 is the first data character following a FLAG character.

Table 7 - CONTROL field

Bit 7	6	5	4	3	2	1	Bit 0
reserved = 0h				FRAME TYPE		FSN or RESET TYPE	

7.1.1 FRAME TYPE field

The FRAME TYPE field defined in Table 8 specifies the format of the frame.

Table 8 - FRAME TYPE field

Frame type	Description
00b	APPLICATION FRAME
01b	RESERVED
10b	PRIVILEGED FRAME
11b	CONTROL FRAME

7.1.2 RESET TYPE field

The RESET TYPE field specifies the type of reset that shall be performed as a result of receiving a valid Control frame as defined in Table 9.

Table 9 - RESET TYPE field

Reset type	Description
00b	LINK RESET
01b	TOTAL RESET
10b	RESERVED
11b	ABSOLUTE RESET

A Link Reset frame is exchanged between two adjacent nodes to report transmission errors and invoke the Link ERP process (see 9.5.1).

A Total Reset frame causes the destination node to re-initialize its internal state. It is used to recover from a catastrophic error (see 9.5.2).

An Absolute Reset frame causes the destination node to perform a Total Reset and optional additional resets. An Absolute Reset frame optionally invokes the POST (see 9.5.2).

7.1.3 FSN field

The FSN field of a Privileged, Reserved or Application frames contains a FRAME SEQUENCE NUMBER (FSN) that protects against frames being lost by a transmission error. For example, if the sole FLAG character between two frames is corrupted into a valid character then the two frames are merged into one.

The FSN increments modulo 4 in each successive Privileged, Reserved, or Application frame.

Each port shall maintain a 2-bit TRANSMIT SEQUENCE NUMBER and shall copy this into the FSN field of each Privileged, Reserved or Application frame sent. The TRANSMIT SEQUENCE NUMBER is set to 00b in the Disabled state. It is incremented modulo 4 when the trailing FLAG character of a Privileged, Reserved or Application frame is transmitted, regardless of whether the frame is acknowledged. However the TRANSMIT SEQUENCE NUMBER is not incremented if a Control frame is transmitted or the transmitter aborts a frame, as described in 6.2.6.

Each port shall maintain a 2-bit RECEIVE SEQUENCE NUMBER. The RECEIVE SEQUENCE NUMBER is set to 00b in the Disabled state. It is incremented modulo 4 when the trailing FLAG character of a valid Privileged, Reserved or Application frame is received. However the RECEIVE SEQUENCE NUMBER shall not be incremented for a Control frame or an aborted frame.

When a port receives a Privileged, Reserved or Application frame it checks the FSN field against the RECEIVE SEQUENCE NUMBER as follows:

a) If the FSN field value = RECEIVE SEQUENCE NUMBER, the frame is accepted and an acknowledgment is returned, provided there is no other error.

b) If the FSN field value ≠ RECEIVE SEQUENCE NUMBER then one or more frames have been lost. The current frame shall not be acknowledged and the Link ERP process is invoked resulting in a Link ERP exit with an ALERT CODE value of SEQUENCE ERROR if no other higher priority error exists (see 10.1).

The TRANSMIT SEQUENCE NUMBER and RECEIVE SEQUENCE NUMBER shall be not incremented for Control frames. The TRANSMIT SEQUENCE NUMBER and RECEIVE SEQUENCE NUMBER are also used to recover from transmission errors (see 10.1.3). Since the TRANSMIT SEQUENCE NUMBER and RECEIVE SEQUENCE NUMBER are independent on each link, the router in a dual port or switch node shall regenerate the FSN field as it forwards a frame.

To facilitate future extensions, dual port and switch nodes shall forward a frame with any bit pattern in the CONTROL field, provided the FSN field value is equal to the RECEIVE SEQUENCE NUMBER for any frames with a FRAME TYPE field value of APPLICATION FRAME, RESERVED or PRIVILEGED FRAME. However the destination port shall invoke the Link ERP process if the FRAME TYPE field is RESERVED or the RESET TYPE field is RESERVED resulting in a Link ERP process exit with an ALERT CODE value of FRAME REJECT ERROR for any reserved bit patterns if no other higher priority error exists (see 10.1).

7.2 STATUS field

The STATUS field begins with the next data character following the CONTROL field in a Link Reset frame and contains the Link Status Byte (see 10.1.2).

7.3 PATH field

The PATH field begins with the next data character following the CONTROL field in Total Reset or Absolute Reset frames. The PATH field is of variable length and consists of a Path component (see 7.4.3).

7.4 ADDRESS field

The ADDRESS field defined in Figure 9 begins with the next data character following the CONTROL field of a Privileged, Reserved or Application frame. The ADDRESS field is of variable length and consists of two components, the Path component and the Channel component (see 7.4.3 and 7.4.4).

Path				Channel

1 Byte

Each byte of address field

Bit 7	6	5	4	3	2	1	Bit 0
EXTEND	INDEX						
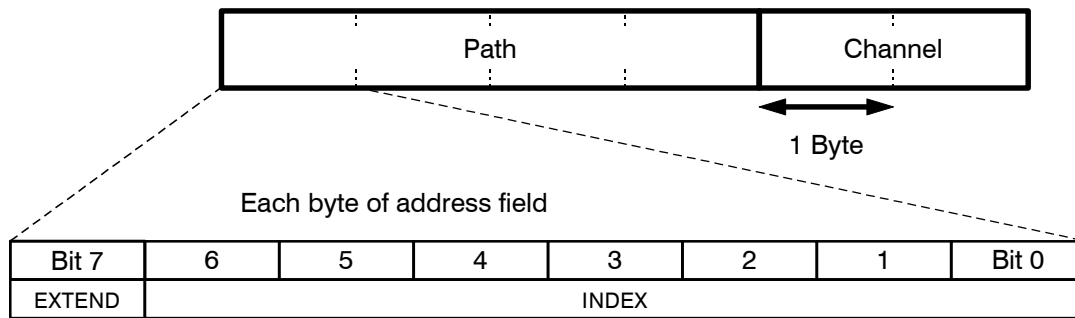
Figure 9 - ADDRESS field

The ADDRESS field in a Privileged, Reserved or Application frame contains from 2 to 6 bytes, depending on the complexity of the Web and the number of Channels implemented by the destination node. It consists of a Path component followed by a Channel component. Each component shall be a minimum of 1 byte and is extended a byte at a time by setting the EXTEND bit.

7.4.1 EXTEND bit

When the EXTEND bit is set in a byte it indicates that the component (Path or Channel) is extended into the following byte. When the EXTEND bit is cleared it indicates that the current byte is the last byte of the component. Thus, within the ADDRESS field, the first byte with the EXTEND bit cleared specifies the end of the Path component and the second byte with the EXTEND bit cleared specifies the end of the Channel component.

7.4.2 INDEX

The INDEX is a 7-bit unsigned integer used by the Path and Channel components.

7.4.3 Path component

The Path component is used to route the frame over a particular path to the destination node. Path addresses are relative to the source node. This simplifies the routing hardware and avoids the need to assign absolute node addresses at power-on.

When a frame is forwarded by the router in a dual port node the first byte of the Path component is decremented. When a frame is forwarded by a switch the first byte of the Path component shall be decremented or deleted according to the rules specified in 8.2.3. If it is deleted the next byte shall be used for subsequent routing decisions.

The length of the Path component is 1 - 4 bytes. The maximum length is not checked explicitly. However, it needs to be specified for other functions (e.g., frame buffers and SMS formats).

7.4.4 Channel component

The Channel component selects a channel within the receiving port at the destination node. A channel consists of the facilities to receive an SMS or to receive a single data transfer. Every port shall provide channel zero to receive SMSs. All other Channels are dynamically allocated.

The transport layer starts and controls data transfers by exchanging SMS frames between the source node and the destination node (see 11.3). The destination node allocates a Channel to receive the Data frames and it indicates the number of bytes that it expects.

The length of the Channel component is 1-2 bytes. The maximum length is not checked explicitly. However, it needs to be specified for other functions (e.g., frame buffers and SMS formats). A single-byte Channel component allows up to 127 data Channels; 2 bytes allow an additional 16,256 data Channels.

7.5 DATA field

The DATA field shall be present in all frames except Control frames. The contents of the DATA field in Privileged SSA-TL1 SMSs is defined in 11.2. The contents of the DATA field in SSA-TL1 Application SMSs is defined in 11.3.

The DATA field starts in the next data character following the ADDRESS field. The destination node locates the start of the DATA field by parsing the ADDRESS field (i.e., the remaining Path byte and the Channel byte(s).)

The DATA field contains from 0 to 128 data characters. The maximum length is defined to ensure compatibility between different nodes in a Web.

Except for the Data frame containing the last byte of a data transfer, the length of the DATA field of any Data frame shall be an integer multiple of 8 bytes. This restriction on the division of a data transfer into frames eases alignment at the parallel interface to the node function. A frame that violates the restriction is optionally rejected by the destination node.

A frame shall contain at most one SMS. The maximum length of the DATA field in an SMS is 32 bytes.

7.6 CRC field

The CRC field consists of the 4 data characters that immediately precede the trailing FLAG character. It is accumulated over the CONTROL, PATH, ADDRESS, LINK STATUS BYTE and DATA fields. The router in dual port and switch nodes may interpret the CONTROL, PATH, and ADDRESS fields as they are received (see 8.1). However, the receiving port shall not assume that any of the fields are valid until the CRC field has been determined to be valid.

The value of the CRC field for a frame shall not be calculated to include any interleaved Special characters.

Each node shall generate the CRC field when sending a frame. Each node shall regenerate the CRC field when forwarding a frame.

7.6.1 Polynomial definitions

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

> NOTE 2 - SSA, FDDI and FCS use the same degree 32 polynomial.

The degree 31 polynomial with all coefficients equal to 1 is $L(x) = x^{31}+x^{30}+x^{29}+x^{28}+...+x^3+x^2+x+1$.

The remainder polynomial produced by the receiver in the absence of errors is $C(x) = x^{32} L(x) \bmod G(x)$
$= x^{31} + x^{30} + x^{26} + x^{25} + x^{24} + x^{18} + x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^9 + x^6 + x^5 + x^4 + x^3 + x + 1$.

The degree k-1 polynomial representing the CONTROL, ADDRESS and DATA fields of the transmitted frame is F(x).

The remainder polynomial is of degree less than 32 is R(x).

The polynomial of degree less than 32 representing the transmitted CRC field is K(x).

The transmitted frame including the CRC field is represented by M(x).

All arithmetic is Modulo 2.

7.6.2 CRC generation

The CRC field is generated with $R(x) = (x^{32} F(x) + x^k L(x))$ mod $G(x)$. This means that the CRC generator shall be preset to all ones at the start of each frame.

The transmitted CRC field is the ones complement of $R(x)$ with $K(x) = L(x) + R(x)$.

The CRC field appended to the transmitted frame is defined by $M(x) = x^{32} F(x) + K(x)$.

7.6.3 Checking

The received frame is divided by $G(x)$. However, to allow for leading zeros the CRC register is first preset to all ones. In the absence of errors the remainder is $C(x) = x^{32} (M(x) + x^k L(x))$ mod $G(x)$. If the CRC remainder is not C704DD7Bh the port shall invoke the Link ERP process with a RECEIVER ERROR field value of CRC ERROR (see 10.1.1.7).

8 Protocol

8.1 Router functions

This clause describes in more detail the functions of the router in a dual port or switch node.

Each node in the path between the source node and destination node forwards the frames with minimum delay achieved by using a cut-through router, rather than store-and-forward. Cut-through routing allows each node to begin forwarding a frame out another port while still receiving characters of the same frame, provided that the out-bound port is not already sending a frame or the WAITING FOR RR FLAG is set.

The minimum latency introduced by a cut-through router is 5 characters since the CRC field is regenerated as a frame is forwarded. The inbound port shall receive the trailing FLAG character before the first CRC character is sent on the outbound port.

> NOTE 3 - In a practical implementation a delay of up to 12 characters is possible. For a string of 16 nodes this allows a minimum delay of less than 10 µs to be achieved at 20 MB/s. With store-and-forward routing the minimum delay is over 100 µs with 128-byte frames.

Dual port nodes shall use a cut-through router to ensure good performance in all applications. Switch nodes may also use a cut-through router.

If a node detects an error in an inbound frame when it has already started to forward the frame, it shall send an ABORT character immediately followed by a FLAG character. The ABORT character causes the destination node to discard the preceding frame without generating a link error. If the destination node has also started to forward the frame then it terminates the forwarding frame with an ABORT character immediately followed by a FLAG character.

8.2 Web routing

Each type of node uses different rules to interpret the Path component in the ADDRESS or PATH fields of an inbound frame. If the receiving port determines that the Path component is invalid according to its rules then it shall invoke the Link ERP process with a RECEIVER ERROR value of FRAME REJECT (see 10.1.1.9).

The following nomenclature is used to define the addressing rules.

    a)   First byte refers to the first byte in the Path or Channel component.

    b)   Second byte refers to the second byte in the Channel component.

    c)   Input port refers to the port of a switch node that receives the frame.

    d)   Output port refers to the selected output port of a switch node.

    e)   Ports refers to the total number of ports on a switch node.

8.2.1 Single port node

A single port node has only one port, and shall interpret the Path component of the ADDRESS or PATH fields as follows:

```
If First byte = 00h then Do;
     Accept the frame;
     Interpret the remainder of the ADDRESS field as a Channel component;
     End;
Else reject the frame;
```

If the first byte of the Path component is non-zero the node shall invoke the Link ERP process with a RECEIVER ERRORS value of FRAME REJECT (see 10.1.1.9).

For error reporting and configuration, the PORT field has a value of one (see 11.2.1 and 11.2.3).

8.2.2 Dual port node

A dual port node has 2 ports and an internal router. Depending on the value of the first byte of the Path component of the ADDRESS or PATH fields, the router either forwards the frame on the out-bound line of the other port, or accepts the inbound frame for processing internally. The router interprets the Path component as follows:

```
If First byte = 00h then Do;
     Accept the frame;
     Interpret the remainder of the ADDRESS field as a Channel component;
     End;
Else If First byte.Index > 0 then Do;
     First byte.Index = First byte.Index - 1;
     Forward the frame via the other port;
     End;
Else reject the frame;
```

If the first byte of the Path component is 80h the node shall invoke the Link ERP process with a RECEIVER ERRORS value of FRAME REJECT (see 10.1.1.9).

For error reporting and configuration the PORT field has a value of one or two (see 11.2.1 and 11.2.3).

8.2.3 Switch node

A switch node shall have an internal router and an even number of ports in the range of 4 to 126 inclusive. If the first byte of the Path component is zero, the switch shall accept the inbound frame from any port for processing internally. The frame is routed out the selected output port. The switch shall use one of two different addressing algorithms, depending on the relationship of the first Path byte to the number of switch ports.

The first addressing scheme shall be identified by the first Path byte index having a value less than or equal to the number of switch ports. An inbound frame shall be forwarded to the output port identified by the first Path byte index. The first Path byte shall then deleted. In this scheme, a first Path byte with the EXTEND bit cleared is a frame reject case, since the last Path byte shall be deleted and the frame forwarded without a Path. This addressing scheme allows for a more flexible port selection, but since the first Path byte is deleted, switches shall only be nested three deep.

The second addressing scheme shall be identified by the first Path byte index having a value greater than the number of switch ports. For routing purposes the switch ports are grouped into pairs. Port 1 is paired with port 2, port 3 is paired with port 4, etc. In general port 2N-1 is paired with port 2N for N = 1, 2, 3, ... 63. An inbound frame shall be forwarded through its paired output port. The first Path byte shall then decremented by the number of switch ports plus one without deleting it. This provides an efficient means to cascade a number of switches.

The switch router interprets the path component of the address field as follows:

```
If First byte = 00h then do;       (*** Process the frame internally ***)
        Accept the frame;
        Interpret the remainder of the ADDRESS field as a Channel component;
     end;
else if (First byte.Index > Ports)
     then do;       (*** Decrement and route out paired port ***)
        First byte.Index = First byte.Index - (Ports + 1);
        Forward the frame via other port of pair;
     end;
     else if (Input port = First byte.Index) or (First byte.Extend = 0b)
          then Reject the frame
          else do;      (*** Delete and route out selected port ***)
            Output port = First byte.Index;
            Delete First byte;
            Forward the frame via Output port;
          end;
If First byte = 80h then reject the frame;
```

The Link ERP process with a RECEIVER ERRORS value of FRAME REJECT (see 10.1.1.9) shall be invoked for the following conditions:

a) First byte = 80h (i.e., the frame is addressed to the switch, but another Path component byte exists);

b) First Byte.index ≤ Ports and Extend = 0b (i.e., the first byte shall be discarded, but no additional bytes exist);

c) or First Byte.index ≤ Ports and First byte.Index = Receiving port (i.e., an attempt has been made to output the frame via the receiving port).

Switch ports shall be numbered sequentially starting at one (see 11.2.1 and 11.2.3).

8.2.4 Routing example

An example of a complex Web is shown in Figure 10. The numbers in the edges of the boxes are the port numbers. For illustration purposes the ports are numbered in a clockwise direction. The number inside each box is the hexadecimal Path address of that node relative to the indicated Configutor node. The dual port nodes on the cyclic path allow two alternative paths from the Configutor node. When a particular path includes a switch it has a shorter alias address if the switch input and output ports are in the same pair.

All Configutor nodes execute the Configuration process (see 9.4.1) to discover the ports that are Operational. A Configutor node also selects a primary Path component to each other node and calculates the corresponding Return Path.

# Indicates an alias path address where
the first byte is decremented by the
number of ports +1 rather than deleted
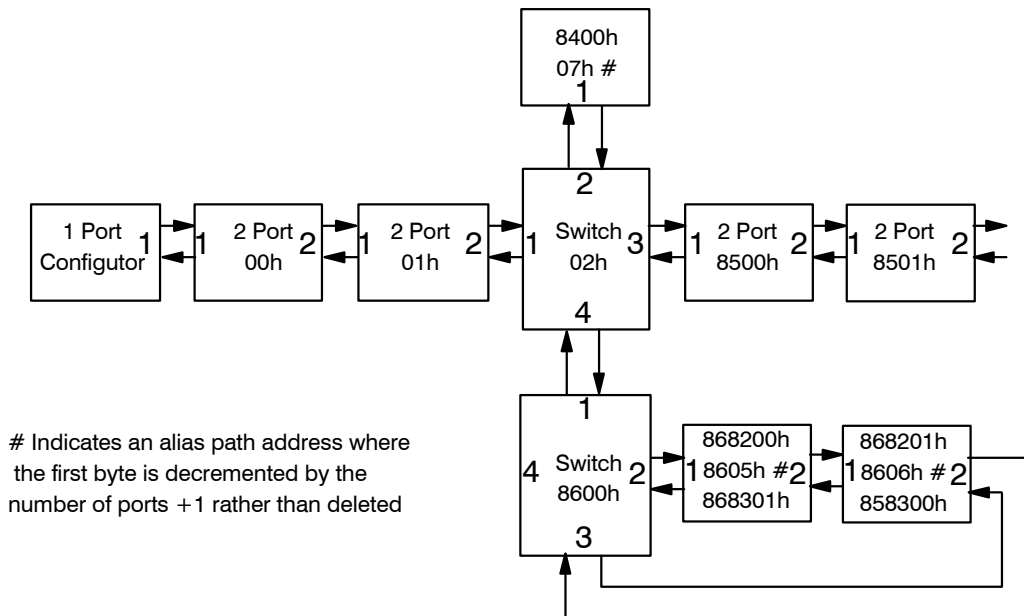
Figure 10 - Path addressing

8.3 Node routing

The Channel component is interpreted by all nodes as shown below.

```
If First byte.Index = 0 then Do;
     If First byte.Extend = 0b then select SMS Channel;
     Else reject the frame;
     End;
Else Do;
     If First byte.Extend = 0b then
          select Data Channel = First byte.Index;
     Else If Second byte.Extend = 0b then
          select Data Channel = (128 * First byte.Index) + Second byte.Index;
     Else reject the frame;
     End;
```

Channels 0-127 are addressed by one byte. Channels 129-16 383 are addressed by two bytes.

If the first byte of the Channel component is 80h or the Channel component is in error, the node shall invoke the Link ERP process with a RECEIVER ERRORS value of FRAME REJECT (see 10.1.1.9).

8.4 Flow control

To implement the necessary flow control, the remote port sends the local port character pairs for each frame it receives:

   a) Acknowledgment is a pair of consecutive ACK characters;
   b) Receiver Ready is a pair of consecutive RR characters.

The Special characters are used in pairs to protect the Acknowledgment or Receiver Ready from being manufactured by transmission errors. A node shall only acts on an Acknowledgment or Receiver Ready pair when it has received both characters of the pair without any other intervening characters.

Flow control operates on each link independently. ACK and RR characters shall not be forwarded by a router.

In full-duplex operation a port may send an Acknowledgment for a received frame or a Receiver Ready if buffer space is available, while it is transmitting another frame. In this case the transmitter gives priority to the

Acknowledgment or Receiver Ready and interleaves it within the frame. This scheme reduces latency and the amount of buffering required by each port to sustain the full data rate of the link.

8.4.1 Acknowledgments

The link protocol requires a port to acknowledge every valid received frame, except for Total Reset, Reserved or Absolute Reset Control frames. A frame is valid if it does not contain any of the errors listed in 10.1.1. The destination port transmits an ACK character pair when it receives a valid frame. When the source port receives the ACK character pair, it shall discard the transmitted frame from its frame buffers.

This protocol guarantees that the transmitter always associates each ACK character pair unambiguously with the corresponding frame independently of propagation delays, the transmission speed and the frame length.

Each port has two associated flags, WAITING FOR ACK FLAG and ACK PENDING FLAG, that control acknowledgments in the order as follows:

a) When a port enters the Disabled state it clears the WAITING FOR ACK FLAG and ACK PENDING FLAG.

b) A port shall set the WAITING FOR ACK flag after it finishes transmitting the trailing FLAG character of any frame except Total Reset, Reserved or Absolute Reset Control frames. A port clears the WAITING FOR ACK FLAG when it receives an ACK character pair. The local port shall discard the corresponding transmit data. The error conditions shall be handled as follows:

   1) After setting the WAITING FOR ACK FLAG a port shall wait to receive the ACK character pair for a minimum of 25 µs and a maximum of 50 µs before setting the ACK bit in the Link Status Byte and invoking the Link ERP process (this range allows a simple binary counter to be used for the timer).

   2) If a port's WAITING FOR ACK FLAG is still set when it finishes transmitting the CRC field of the next frame then it shall not transmit the trailing FLAG character. Instead it sends NUL characters until either the ACK character pair is received or an ACK time-out occurs. If an ACK time-out occurs in this state, then the port shall abort the frame (see 6.2.6), set the ACK bit in the Link Status Byte and invoke the Link ERP process.

   3) If a port receives an ACK character pair when the WAITING FOR ACK flag is cleared, or it receives only a single ACK character, then it shall set the PROTOCOL ERROR value in the RECEIVER ERRORS field of the Link Status Byte and invokes the Link ERP process.

c) A port sets the ACK PENDING FLAG when it receives the trailing FLAG character of a valid frame in either of the following cases:

   1) the port is in the Ready state and the frame is not a Total Reset, Reserved or Absolute Reset frame;

   2) the port is in the Check state and the frame is a Link Reset frame.

d) When the ACK PENDING FLAG is set the port shall transmit an ACK character pair as soon as possible. The ACK PENDING flag is cleared when the ACK character pair has been transmitted.

8.4.2 Acknowledgment example

The timing requirements for acknowledgments and the use of NUL characters is illustrated in Figure 11, where port 1 and port 2 are on opposite ends of a link (i.e., on different nodes). Port 1 finishes sending a first frame and then it immediately sends a second short frame. Due to the delay before port 2 acknowledges the first frame port 1 has to insert a NUL character in the second frame. In order to guarantee that port 1 does not indicate an ACK TIME-OUT error for the first frame, the delay from port 1 setting the WAITING FOR ACK FLAG on port 1 to receiving the second ACK character shall be less than 25 µs. This limits the maximum operating distance of the link, as described in SSA-PH.

Pd - Line propagation delay
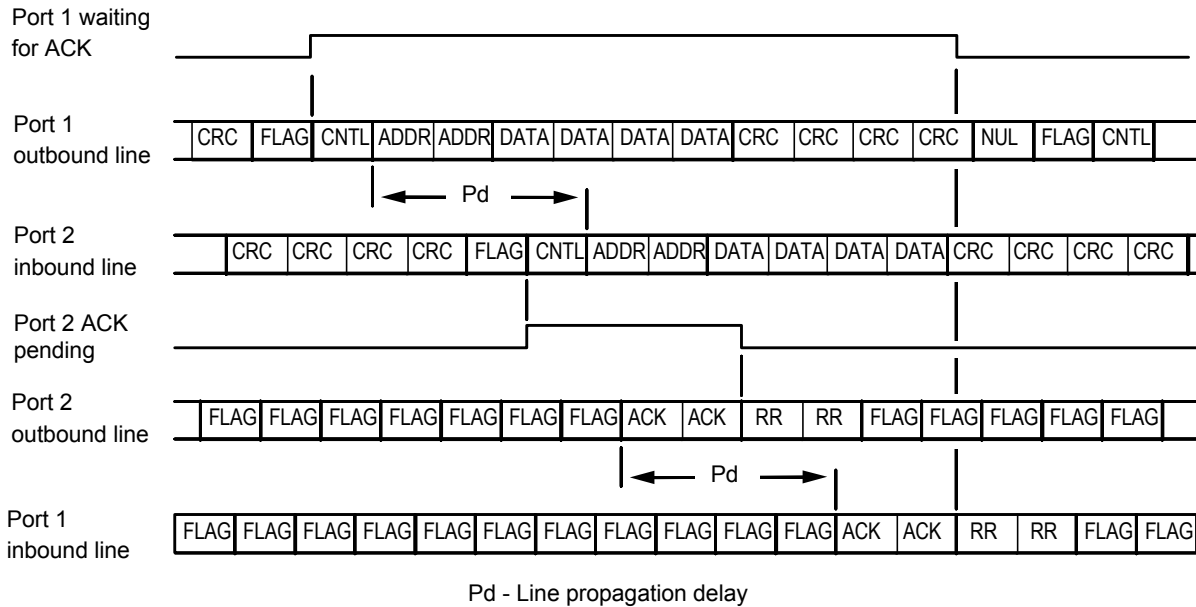
Figure 11 - Timing of acknowledgments

8.4.3 Receiver Ready Pacing

Pacing ensures that a transmitting port does not overrun the available buffer space in the receiving port. The unit of pacing is a frame. The receiver shall have buffer space for at least 1 frame, although additional buffer space is required to sustain the full data rate of the link.

Each port has two associated flags, the WAITING FOR RR FLAG and the RR PENDING FLAG, that control pacing in the order as follows:

a) When a port enters the Enabled state it sets the WAITING FOR RR FLAG and RR PENDING FLAG.

b) A port may start sending a frame when either of the following conditions is satisfied:

1) the port is in the Ready state and it's WAITING FOR RR FLAG is cleared;

2) the transmitted frame is a Control frame and the port is in the Enabled, Check or Ready states.

c) The WAITING FOR RR FLAG is set when a port transmits the CONTROL field of a Privileged, Reserved or Application frame and is cleared when the port receives an RR character pair.

d) When all of the following conditions are satisfied, a port shall transmit an RR character pair after the current character:

1) the port is in the Ready state and the RR PENDING FLAG is set;

2) buffer space is available to receive at least one frame on any active Channel, or shall be available before an incoming frame is received.

e) The RR PENDING FLAG is set when a port receives the CONTROL field of Privileged, Reserved or Application frame. The RR PENDING FLAG is cleared when the port transmits an RR character pair.

8.5 SAT

The origination of Application frames for strings and loops is controlled by the SAT algorithm[2] where SAT refers to Satisfied. The SAT algorithm helps balance between spatial reuse of the individual links and any node being starved of bandwidth.

8.5.1 Terminology

Some SAT terms are defined in the following list.

a) A QUOTA field in the CONFIGURE PORT SMS (see 11.2.3) is used to determine how many frames a port may originate while holding cut through frames.

b) B QUOTA field in the CONFIGURE PORT SMS is used to determine the number of frames a port may originate if no cut through traffic exists (i.e., the link is idle with two consecutive FLAG characters).

c) REFLECT bit in the CONFIGURE PORT SMS is one of the factors to instruct the port to reflect SAT and SAT' characters, breaking the loop or string into two SAT regions (see 8.5.5).

d) REFLECTION FLAG is a node function that determines whether all ports of a node reflect or forward the SAT and SAT' characters (see 8.5.2).

e) Hold quota is a variable containing the A QUOTA field value times the number of current outstanding data transfers from the port. This dynamic scaling of the A QUOTA ensures fairness when a port is originating several outbound data transfers concurrently. The number of frames sent does not have to be balanced between each of the concurrent data transfers.

f) Idle quota is a variable containing the B QUOTA field value times the number of current outstanding data transfers from the port. This dynamic scaling of the B QUOTA ensures fairness when a port is originating several outbound data transfers concurrently. The number of frames sent does not have to be balanced between each of the concurrent data transfers.

g) FRAME COUNTER is a counter that counts the number of Application frames originated since the last time the SAT character was forwarded or reflected by the port. Forwarding or reflecting a SAT character sets the FRAME COUNTER to zero.

h) Originated means that the trailing flag of an originating frame has been sent. Aborted frames are not considered Originated and thus not counted against the SAT quotas.

i) Satisfied refers to the condition of a port whereby either of the two following conditions is met.

1) The port has originated at least Hold quota number of frames since it previously forwarded or reflected a SAT character.

2) The port has no frames to originate when holding the SAT character.

8.5.2 Reflection versus forwarding

If the REFLECTION FLAG is cleared, all port(s) of a node shall forward SAT and SAT' characters. If the REFLECTION FLAG is set, all port(s) of a node shall reflect SAT and SAT' characters.

A port shall forward a SAT character after receiving a SAT character by holding it according to the rules of the SAT algorithm and then generating a SAT character on the outbound lines of the port that did not receive the SAT character. A dual port node shall unconditionally forward a SAT' character after receiving a SAT' character by generating a SAT' character on the outbound lines of the port that did not receive the SAT' character.

A port shall reflect a SAT character after receiving a SAT character by holding it according to the rules of the SAT algorithm and then generating a SAT' character on the outbound lines of the same port that received the SAT character. A SAT' character is reflected unconditionally by generating a SAT character on the outbound lines of the same port that received the SAT' character.

---

[2]For Informative reference only. I. Cidon and Y. Ofek, "MetaRing: A Full-Duplex Ring with Fairness and Spatial Reuse", IEEE INFOCOM '90, pp. 969-981.

8.5.3 The SAT algorithm

The node at the end of a string shall be a single port node, a dual port node with one port not operational or a switch (a dual port node with both ports operational is not at the end of a string by definition). Thus all nodes shall implement the SAT algorithm.

The SAT algorithm is described initially for a loop of dual port nodes, with enhancements for a string shown in 8.5.4. Application frames entering a loop are controlled by a SAT character that circulates in the opposite direction to the frames that it controls, as shown in Figure 12. Circulating the SAT character in the opposite direction maximizes spatial reuse.

The SAT algorithm is applied independently to clockwise and counter-clockwise traffic. Therefore there are two SAT characters circulating on the loop, one in each direction.
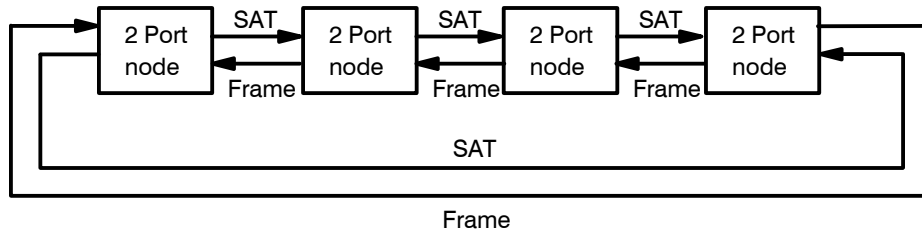
Figure 12 - SAT operation for a loop

When a port receives a SAT character it forwards or reflects the character via the other port to the next node if it is Satisfied. Otherwise the port holds the character until it is Satisfied, even if it is waiting for RR. Then it forwards or reflects the SAT character and sets the FRAME COUNTER to zero.

Privileged and Control frames are not controlled by the SAT algorithm and, subject to the link protocol, they may be sent at any time. A port may originate an Application frame, subject to the link protocol, under either of the following conditions.

   a) The port is holding the SAT character and the port is not Satisfied.

   b) There is no frame waiting to be forwarded from another port, and Idle quota is greater than FRAME COUNTER.

The ability of a port to originate frames is broken into two cases (a and b), and one case (c) not able to originate frames.

   a) FRAME COUNTER < Hold quota.
       Before the SAT character arrives, the port shall originate frames only when there are no cut through frames pending. After the SAT character arrives, the port shall hold cut through frames until Satisfied, then the SAT character is forwarded or reflected and the FRAME COUNTER shall be set to zero.

   b) FRAME COUNTER ≥ Hold quota and FRAME COUNTER < Idle quota.
       Before the SAT character arrives, the port shall originate frames only when there are no cut through frames pending. When the SAT character is received, it shall be forwarded or reflected and the FRAME COUNTER shall be set to zero.

   c) FRAME COUNTER ≥ Idle quota.
       The port shall not originate any frames. When the SAT character is received, it shall be forwarded or reflected, the FRAME COUNTER shall be set to zero and Application frames may be originated according to the SAT algorithm.

Correct operation of the SAT algorithm requires B QUOTA ≥ A QUOTA, otherwise a node is never Satisfied and it does not forward or reflect the SAT character.

A port shall be able to receive SAT and SAT' characters while it is in any state, provided that character synchronization is maintained. However the port shall only transmit SAT or SAT' characters when in the Ready state.

8.5.4 Extensions of the SAT algorithm for strings

The extensions to the algorithm for a string are as follows:

a) If a port at the end of a string receives a SAT character then it reflects the SAT character as a SAT' character traveling in the opposite direction after it is Satisfied (see Figure 13).

b) A dual port node with both ports operational forwards SAT' received from either port unconditionally.

If a port at the end of a string receives a SAT' character then it reflects the SAT' character as a SAT character traveling in the opposite direction immediately without resetting the FRAME COUNTER
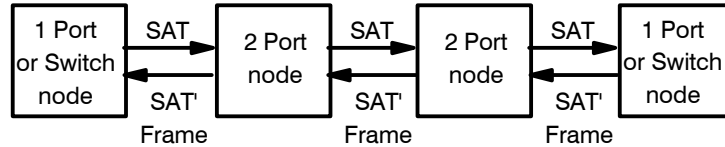
Figure 13 - SAT operation for a string

The reflection of SAT as SAT' characters and SAT' as SAT characters is dynamic. If a string of dual port nodes is connected into a closed loop then SAT' characters are no longer generated. To avoid a previous SAT' character circulating indefinitely a dual port node with both ports operational shall discard SAT' characters as follows:

a) a SAT' character received on port 1 is discarded if no SAT character has been received on port 2 since the previous SAT' character was received on port 1;

b) a SAT' character received on port 2 is discarded if no SAT character has been received on port 1 since the previous SAT' character was received on port 2.

8.5.5 SAT regions

In some applications where each Configutor node talks exclusively with a subset of the Responders, the traffic on a string or loop is naturally divided into separate regions, as illustrated in Figure 14. In this example, significant traffic is only carried between the following nodes: Configutor1 - Responder1, Configutor1 - Responder2, Configutor2 - Responder3 and Configutor2 - Responder4.
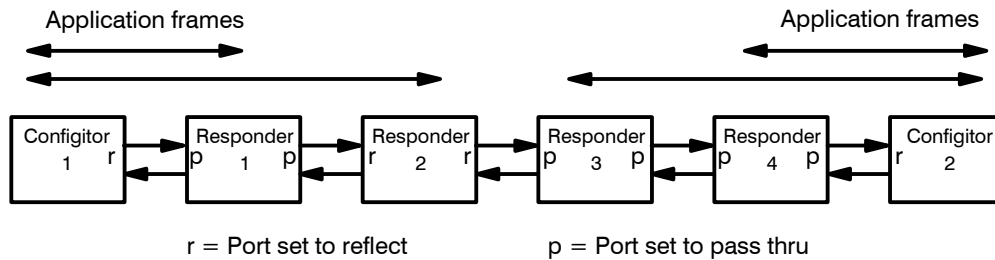
Figure 14 - Web with 2 SAT regions

In this application, any trade-off between fairness and spatial reuse is avoided by logically dividing the Web into 2 regions, each with its own SAT algorithm. This is achieved by setting the REFLECT bit in the CONFIGURE PORT SMSs for either port on Responder 2 (or Responder 3). The ports on Responder 2 then reflects a SAT character as a SAT' character and a SAT' character as a SAT character. Responder 2 still behaves as a dual port node for frame routing (e.g. to allow each system to check the operation of the other).

This facility allows 100% spatial reuse between regions and perfect fairness in each region if A QUOTA=B QUOTA.

If one Configutor node fails the remaining system takes over all of the Responders after issuing a CONFIGURE PORT SMS with new REFLECT, A QUOTA, and B QUOTA fields (if needed).

8.5.6 Timing requirements of the SAT algorithm

All ports incorporate a SAT TIME-OUT timer to detect loss of the SAT character due to a link error. The SAT TIME-OUT timer is set to zero when the port forwards a SAT character or reflects a SAT character as a SAT' character. If the SAT TIME-OUT timer counts 100 ms, then a SAT character is created and treated as if it has been received. Multiple SAT characters circulating in the same direction results if several ports generate new SAT characters simultaneously. However the fairness algorithm still operates correctly and the characters eventually merge into a single character as described below.

If a port receives a SAT character while it is holding the previous SAT then it discards the second SAT character.

Each of the following functions shall not introduce a delay exceeding 10 character periods per node:
   a) Forwarding a SAT character when the port is Satisfied, subject to the delay described in the paragraph below.
   b) Reflecting a SAT character as a SAT' character when the port is Satisfied.
   c) Forwarding a SAT' character.
   d) Reflecting a SAT' character as a SAT character.

For a loop or string with only a few nodes the SAT character rotates very rapidly if all nodes are Satisfied. This impacts frames being transmitted in the other direction. Therefore ports shall perform the following two delays.
   a) A port shall not forward a SAT character until at least 100 character periods have elapsed since it forwarded the previous SAT character.
   b) A port shall not reflect a SAT character as a SAT' character until at least 100 character periods have elapsed since it reflected the previous SAT character.

8.5.7 Configuring the quotas

The A QUOTA and B QUOTA values for each port are allocated by the Master node in a CONFIGURE PORT SMS. The optimum values depend on the total number of nodes in each string or loop and the traffic pattern. This clause describes how the values are determined.

The SAT ROTATION time is the time taken by the SAT character to circulate once around a string or loop, as measured in character periods. SAT ROTATION time is a minimum when all ports are Satisfied and they forward the SAT characters immediately (SAT' characters are always forwarded immediately). SAT ROTATION time increases when the Web becomes heavily loaded and ports hold the SAT character.

The SAT DELAY time is the minimum time to propagate or reflect SAT or SAT' characters through a single node whose inbound port is Satisfied, as measured in character periods.

For a string with N nodes the minimum SAT ROTATION time is 2*N*SAT DELAY characters. The corresponding value for a loop is only N*SAT DELAY characters since the propagation delay for SAT' character is eliminated. SAT ROTATION time is always greater than 100 character periods. However after a link failure the loop becomes a string and the minimum SAT ROTATION time doubles. Therefore it is advisable to always allow for a minimum SAT ROTATION time of 2*N*SAT DELAY time.

The performance advantages of SSA are due in part to its ability to frame-multiplex SMSs and data with low latency. For optimum frame multiplexing and spatial reuse the SAT ROTATION time is minimized. This applies both under light load (when most nodes are Satisfied) and under heavy load (when nodes hold the SAT character).

For long strings SAT DELAY time is a major contributor to SAT ROTATION. Consequently SAT DELAY time has been specified as 10 characters maximum. The SAT and SAT' characters have a high priority for transmission (see 8.6).

The FRAME LENGTH is the total length of a frame. This is typically 136 characters for a data frame. To avoid throttling when a single node wants to originate data frames, B QUOTA $\geq$ (SAT ROTATION time/FRAME LENGTH) $\geq$ (2*N*SAT DELAY time/FRAME LENGTH) For the extreme case of a string with 129 nodes this requires a minimum B QUOTA of 19 frames.

In general there is a compromise between fairness and spatial reuse. Setting B QUOTA=A QUOTA guarantees perfect fairness but it limits spatial reuse. Choosing B QUOTA»A QUOTA permits maximum spatial reuse and therefore maximum total through-put. However in some scenarios this favors the active node that is furthest upstream since cut through traffic normally has priority. Therefore the upstream node is able to originate B QUOTA frames for each rotation of the SAT character, whereas each down-stream node only originates A QUOTA frames. A reasonable compromise for most applications is B QUOTA≈4*A QUOTA.

8.6 Priority for transmission

When a port has multiple characters or frames eligible for transmission, the priority order shall be as follows:

a) (Highest priority) A FLAG character if 10 consecutive FLAG characters have not yet been transmitted after entering the Ready state.

b) Originate the second character of a pair of protocol characters (i.e., ACK-ACK, RR-RR, or ABORT-FLAG).

c) Transmit any pending SAT or SAT' character(s).

d) Transmit any pending User Defined character(s).

e) Originate an ACK character pair.

f) Originate an RR character pair.

g) Continue transmitting the current frame (Including NUL and ABORT characters).

h) Forward a Total Reset, Reserved or Absolute Reset Control frame.

i) Originate a Control frame.

j) Transmit an Application or Privileged frame. Dual port nodes normally give priority to frames being forwarded from the other port. This reduces the latency for long strings and loops. However when the transmitter in a dual port node is holding the SAT token it gives priority to frames being originated by the node itself. This improves fairness by reducing the SAT rotation time (see 8.5.7).

NOTE 4 - It is recommended that switch nodes ensure equal fairness between frames being forwarded by the node itself, the other port of a pair and by other ports.

k) (Lowest priority) Transmit FLAG or DIS characters when idle.

Note that each item in this list has associated pre-requisite conditions before it is eligible for transmission (e.g., Application and Privileged frames cannot be transmitted if the WAITING FOR RR FLAG is set).

9 Link management

9.1 Port attributes

The following clauses describe those attributes associated with a port.

9.1.1 Frame buffers

Since the unit of pacing is a frame each port shall provide buffering to receive at least 1 frame. The buffer shall accommodate the maximum-length CONTROL, ADDRESS and DATA fields (i.e., 1+6+128 = 135 bytes total).

In certain occurrences of a transmission error the Link ERP process retransmits the last one or two frames. Therefore a port shall also retain each frame it transmits until it receives the corresponding ACK character pair.

To achieve continuous communication at the full bandwidth of the link it is necessary for each port to have additional buffering. If a port has a pair of transmit buffers available and a pair of receive buffers to provide A/B buffering, one buffer of each pair is emptied/filled by the link while the other is filled/emptied by the application.

NOTE 5 - Additional dedicated buffers are optional for SMSs to avoid impacting data transfers due to SMS processing delays.

The number and type of buffers in a port are vendor specific. However the following requirements shall be met to ensure performance and compatibility with other nodes in a Web.

  a) All ports shall be able to sustain data transfers with 128 byte DATA fields at the full bandwidth of the link.

  b) When there is no contention, the Router in dual port or switch nodes shall be able to forward the maximum length frames continuously.

  c) All ports shall have an additional buffer to hold a Control frame when all other buffers are full.

9.1.2 TRANSMIT POINTER and RETRY POINTER

The transmit buffering shall be managed to allow correct recovery after a link error. In the case of recovered errors, the Link ERP process retransmits or discards the last one or two frames that were transmitted before an error. Each port shall maintain sufficient status to identify the buffers containing these frames and the order that they were transmitted.

For example, if there are N transmit buffers and the transmitter always accesses them in a cyclic sequence then the following two pointers provide sufficient information:

  a) The TRANSMIT POINTER points to the buffer that is to be transmitted next. It is incremented by 1 modulo N each time the transmitter sends a trailing FLAG character without a preceding ABORT character.

  b) The RETRY POINTER points to the next buffer to be acknowledged. It is incremented by 1 modulo N each time an ACK character pair is received while the WAITING FOR ACK FLAG is set. Normally the RETRY POINTER follows the TRANSMIT POINTER closely but when an error occurs it is possible to lag by 1.

The above scheme is an example used later in the description of the Link ERP process. However, sometimes it is impossible to guarantee cyclic use (e.g., if some buffers are reserved for SMSs). In this case, an alternate solution is to keep a log of the buffers that have been transmitted but not acknowledged, and the appropriate actions to be taken when TRANSMIT POINTER and RETRY POINTER are referenced.

9.1.3 Port states

Each port shall be in one of four states as shown in Figure 15. The current state is inspected by the node processor. State changes are controlled by the node processor or change automatically when certain events occur.

Power On, Total Reset, Absolute Reset

Disabled

Tx sends DIS characters

Rx looks for character synchronization

Rx only recognizes Total Reset, Reserved and Absolute Reset Control frames

Beginning Communication process

Enabled

Tx sends FLAG characters
Rx looks for FLAG characters

Tx or Rx detects line fault

Rx decodes FLAG character

Ready

Normal operation

Rx or Tx detects link error
Rx detects DIS or Link Reset

Check

Tx sends only Control frames
Tx ACKs Link reset only
Rx processes control frames only
Rx processes ACK but not RR characters

Link ERP process disables port

Figure 15 - State transitions

9.1.3.1 Disabled state

The Disabled state is entered after power on, after processing an Absolute Reset or Total Reset frame before the link is made Operational, or is entered during the Link ERP process. Upon entering the Disabled state, the following port functions are initialized:

a) the LINK RESET FLAG is cleared;

b) the TRANSMIT SEQUENCE NUMBER is set to 00b;

c) the RECEIVE TRANSMIT NUMBER is set to 00b;

d) the RR PENDING FLAG and WAITING FOR RR FLAG are set;

e) the ACK PENDING FLAG and WAITING FOR ACK FLAG are cleared;

f) and all of the error latches are cleared.

The port functions that are not initialized include the following:

a) the Frame buffers;

b) the TRANSMIT POINTER;

c) the RETRY POINTER;

d) the port mode;

e) the Port table;

f) the Queued Asynchronous Alert table;

g) and the Asynchronous Alert Address table.

In the Disabled state the transmitter sends DIS characters, maintaining DC balance and allows the remote receiver to maintain character synchronization during the Link ERP process. The receiver attempts to acquire character synchronization. If synchronization is achieved the node responds only to a Total Reset, Reserved or Absolute Reset Control frame.

The minimum duration of the Disabled state is 200 character periods. This allows sufficient time to acquire character synchronization when beginning communication since the remote port cannot enter the Ready state until the local port changes to the Enabled state and sends FLAG characters. It also guarantees that the remote node recognizes the Disabled state during the Link ERP process.

The LINK RESET RECEIVED FLAG shall be cleared in the Disabled state(see 9.1.9).

9.1.3.2 Enabled state

The Enabled state is a transient state that is entered from the Disabled state prior to entering the Ready state.

When a port is ready to begin communications the node processor changes the port to the Enabled state (see 9.5). In the Enabled state the transmitter shall transmit FLAG characters and the receiver shall look for FLAG characters. When the receiver decodes a FLAG character the port shall enter the Ready state.

9.1.3.3 Ready state

The Ready state is the state where normal transmission and reception of frames occurs.

In order to allow the remote port sufficient time to enter the Ready state, after a port enters the Ready state it shall transmit at least 10 FLAG characters before sending any other character.

9.1.3.4 Check state

The Check state is entered automatically from the Ready state when a port detects an error or it receives a Link Reset frame. It shall also be entered from the Enabled state when the line driver/receiver detects a line fault. The transition to the Check state invokes the Link ERP process. Except for Control frames, the link is inoperable until the port returns to the Ready state.

When a port enters the Check state the transmitter stops sending Privileged, Reserved, and Application frames after completing or aborting the current frame, if any. The transmitter then sends FLAG characters continuously, except in the following cases:

    a)   acknowledge a Link Reset frame;

    b)   send a Link Reset frame;

    c)   send a User Defined character;

    d)   or Abort a Link Reset frame.

When a port is in the Check state the receiver shall discard any incoming Privileged, Reserved or Application frames without generating an ACK or RR character pair. The receiver shall discard RR character pairs but ACK character pairs shall be accepted and acted upon.

9.1.4 Port modes

Independent of the state (see 9.1.3) a port operates in one of three modes, as described in the following clauses.

9.1.4.1 Normal mode

Subject to the port state, Normal mode allows any type of frame to be sent and received.

Normal mode is entered from Privileged mode when the port processes a CONFIGURE PORT SMS with a MODE field value of SET NORMAL MODE to the port (see 11.2.3). The Master node shall be capable of selecting Normal mode internally.

9.1.4.2 Privileged mode

In Privileged mode, a port may transmit and receive Control and Privileged frames and shall discard Application and Reserved frames. When the transmitter discards a frame it shall not increment the TRANSMIT SEQUENCE NUMBER. The receiver discards Application and Reserved frames. When the receiver discards a frame, the port shall generate ACK and RR character pairs normally and shall increment the RECEIVE SEQUENCE NUMBER.

Privileged mode is entered from Wrap mode on exit from the power-on self-test (POST) and from Normal mode when the Link ERP process exits with an ALERT CODE value. Privileged mode is entered from any mode after a Total Reset, Reserved or Absolute Reset Control frame is addressed to the node. Privileged mode is entered when the Master sends a CONFIGURE PORT SMS with a MODE field value of SET PRIVILEGED MODE. The Master shall be capable of selecting Privileged mode internally.

9.1.4.3 Wrap mode

If the port is in Wrap mode, the transmitter output of a port shall be internally connected to the receiver input of the same port. In Wrap mode, frames and their responses share the same line. Except for the line driver and receiver, Wrap mode allows a port to be fully tested without needing a remote node. This is useful to test the local hardware during the POST.

When the port is in Wrap mode, the outbound line shall be held at logic zero and any signal on the inbound line shall be ignored. The output of the line fault detector is ignored.

The port shall be placed in Wrap mode at power-on. On exit from the POST, the port enters Privileged mode, Disabled state and the Beginning Communications process is started. The port is placed in the Wrap mode when the Master sends a CONFIGURE PORT SMS with a MODE field value of SET WRAP MODE. The Master shall be capable of entering and exiting the Wrap mode internally.

When the local port enters the Wrap mode, the remote port at the other end of the link invokes the Link ERP process. The Link ERP process exits unsuccessfully and the remote port optionally invokes the Asynchronous Alert process, depending on how the remote port was configured.

When a port is in the Wrap mode, the OPERATIONAL FLAG shall be cleared. If a received frame is addressed to the port it is discarded and the receiving port invokes an Asynchronous Alert process.

Care shall be taken when placing a port in Wrap mode anytime after the POST since in some configurations if the node processor hangs it is impossible to reset it without a power off-on sequence.

> NOTE 6 - Optionally, a node detects when it is not receiving characters on any of its ports and activates a Service indicator (typically an LED). The Master activates the Service indicator on a failing node by wrapping all ports on the remote nodes that are connected to it (the Master node issues CONFIGURE PORT SMSs to the remote nodes or it wraps its own ports internally).

9.1.5 OPERATIONAL FLAG

The OPERATIONAL FLAG is set if the port is able to queue a Privileged frame for transmission to the remote node.

The OPERATIONAL FLAG is cleared after a power-on reset. A port sets the OPERATIONAL FLAG when it enters the Ready state and is not in Wrap mode. When a port changes the OPERATIONAL FLAG from cleared to set, it shall invoke the Asynchronous Alert process with an ALERT CODE value of PORT NOW OPERATIONAL.

The OPERATIONAL FLAG is cleared when the Link ERP process exits, the port is placed in the Wrap mode or the port is reset. The OPERATIONAL FLAG remains set while the port is performing the Link ERP process or when the Link ERP process completes successfully.

The configuration process uses the contents of the QUERY NODE REPLY SMS to determine whether each port of the addressed node has the OPERATIONAL FLAG set or cleared.

A node shall not originate any frame other than a Total Reset, Reserved or an Absolute Reset Control frame via a port that has the OPERATIONAL FLAG cleared.

The reception of a frame, (whether the frame is Acknowledged or not) does not depend on the receiving port's OPERATIONAL FLAG . For example, an Application frame is received when the port is in the Ready state, in Normal mode and the RR PENDING FLAG is cleared.

The router in a dual port or switch node forwards frames as follows:

a) An Application or Privileged frame is only forwarded if the addressed port's OPERATIONAL FLAG is set. Otherwise the node invokes the Asynchronous Alert process with an ALERT CODE value of ADDRESSED PORT NOT OPERATIONAL (see 11.2.1), and the router discards the frame. The receiving port acknowledges the frame and remains in the Ready state.

b) A Total Reset, Reserved or Absolute Reset Control frame is forwarded regardless of the state of the addressed port's OPERATIONAL FLAG.

9.1.6 TRANSMIT SEQUENCE NUMBER and RECEIVE SEQUENCE NUMBER

The port shall maintain two 2-bit counters, the TRANSMIT SEQUENCE NUMBER and the RECEIVE SEQUENCE NUMBER, to detect lost frames (see 7.1.3).

9.1.7 WAITING FOR RR FLAG and RR PENDING FLAG

The port shall maintain two flags, WAITING FOR RR FLAG and RR PENDING FLAG, to control the pacing of frames (see 8.4.3).

9.1.8 WAITING FOR ACK FLAG and ACK PENDING FLAG

The port maintains two flags, WAITING FOR ACK FLAG and ACK PENDING FLAG, to control the acknowledgment of frames (see 8.4.1).

9.1.9 LINK RESET RECEIVED FLAG

The LINK RESET RECEIVED FLAG shall be used during the Link ERP process to determine if the process was entered due to a link error (cleared) or due to receiving a Link Reset frame (set). The LINK RESET RECEIVED FLAG shall be cleared when the port enters the Disabled state. The LINK RESET RECEIVED FLAG shall be set when the port receives a Link Reset frame and is not in the Disabled state. The LINK RESET RECEIVED FLAG is tested during the Link ERP process.

9.1.10 Port table

The node is responsible for maintaining the information necessary to Invoke the Asynchronous Alert process. The information is supplied by the CONFIGURE PORT SMS on a per port basis.

The format of the Port table is vendor specific. One possible implementation contains the fields defined in Table 10. There is one entry for each port.

Table 10 - Port table entry

Field	Description
AA VALID FLAG	A flag indicating if the port table entry is valid
AA PORT	The port to be used for the ASYNC ALERT SMS
AA RETURN PATH	The path to be used for the ASYNC ALERT SMS frame
AA TAG	The TAG field value to be used for the ASYNC ALERT SMS

When the node powers up, the Port table shall be initialized with an entry for each port and with the AA VALID FLAG cleared. When a CONFIGURE PORT SMS is received, the Port table entry associated with the PORT field specified in the CONFIGURE PORT SMS is set as follows:

a) The AA VALID FLAG is set.

b) The port number the CONFIGURE PORT SMS was received on is loaded into AA PORT field.

c) The RETURN PATH field from the CONFIGURE PORT SMS is loaded into AA RETURN PATH field.

d) The TAG field from the CONFIGURE PORT SMS is loaded into AA TAG field.

9.1.11 Queued Asynchronous Alert table

The node is responsible for maintaining the information necessary for queuing Asynchronous Alerts.

The format of the Queued Asynchronous Alert table is vendor specific. One possible implementation contains the fields defined in Table 11. Each entry contains a queued Asynchronous Alert.

Table 11 - Queued asynchronous alert table entry

Field	Description
AA ALERT CODE	A three byte value consisting of the ASYNC ALERT SMS ALERT CODE value
AA CONTROL	A 1 byte field containing the CONTROL field of the frame responsible for the Asynchronous Alert. It is zero if not applicable.
AA CHANNEL	A 2 byte field containing the CHANNEL field of the frame responsible for the Asynchronous Alert. It is zero if not applicable.
AA FRAME	A 21 bytes field containing the first 21 bytes of the associated frame's DATA field, left justified and zero filled.

A separate Queued Asynchronous Alert table is maintained for each port, containing an entry for each Asynchronous Alert to be reported against that port. The port to be used to report the Asynchronous Alert is found in the Port table (see 9.1.10).

9.1.12 Asynchronous Alert Address table

The Master is responsible for maintaining the information necessary for associating ASYNC ALERT SMSs with a particular node.

The format of the Asynchronous Alert Address table is vendor specific. One possible implementation contains the fields defined in Table 12. The table exists for each port and contains an entry for each valid TAG value used in the last CONFIGURE PORT SMS for a given node.

Table 12 - Asynchronous alert address table entry

Field	Description
AA TAG	A two byte value consisting of the Asynchronous Alert TAG value
AA ADDRESS	A four byte value consisting of value to be used for the Path component of the ADDRESS field of any ASYNC REPLY SMS sent to the node.
AA PORT	A one byte value indicating the port to use to send the ASYNC REPLY SMS.
AA NODE IDENTIFIER	A pointer into the Configuration table to associate the Asynchronous Alert with a particular node.

9.1.13 SAT TIME-OUT TIMER

The SAT TIME-OUT TIMER shall be implemented by all ports and used to determine that a SAT character has been lost. The SAT TIME-OUT TIMER is used in the SAT algorithm (see 8.5.3) and is a 100 ms timer.

The node shall activate the SAT TIME-OUT TIMER when the port completes Power On, Total Reset, or Absolute Reset. The port shall restart the SAT TIME-OUT TIMER when the port forwards or reflects a SAT character. When the SAT TIME-OUT TIMER expires, the port shall create a SAT character and treats it as if it has been received (see 8.5.6).

9.2 Node based data structures

9.2.1 Unique ID

Each SSA node shall have a Unique ID that is globally unique. A UNIQUE ID field is an 8-byte field, comprised of two reserved bytes (containing zero) followed by a 6-byte IEEE Universal Address. The first 3 bytes of the IEEE

Universal Address are an Organizationally Unique Identifier (OUI) that is assigned by IEEE[3]. The last 3 bytes are a product identifier that is administered by the OUI assignee.

As an example, the Organizationally Unique Identifier AC-DE-48 and the product identifier 000080h shall generate the Unique ID 0000.ACDE.4800.0080h.

The UNIQUE ID defined in Table 13 is used in the configuration process to detect cyclic paths and with the QUIESCE SMS to handle certain link errors. The UNIQUE ID is used to identify a particular node when the Web topology changes.

Table 13 - UNIQUE ID field

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	reserved							
1	reserved							
2	IEEE ORGANIZATIONALLY UNIQUE IDENTIFIER							
3	IEEE ORGANIZATIONALLY UNIQUE IDENTIFIER							
4	IEEE ORGANIZATIONALLY UNIQUE IDENTIFIER							
5	PRODUCT IDENTIFIER							
6	PRODUCT IDENTIFIER							
7	PRODUCT IDENTIFIER							

9.2.2 Master Asynchronous Alert table

The Master shall maintain the information necessary for associating ASYNC ALERT SMSs with a particular node. The Master Asynchronous Alert table keeps track of this information.

The format of the Asynchronous Alert Address table is vendor specific. One possible implementation contains the fields defined in Table 14. A table entry exists for each tag value given for an ASYNC ALERT SMS. When the Master issues a CONFIGURE PORT SMS, an entry is created or updated in the Master Asynchronous Alert table. When an ASYNC ALERT SMS is received, the table is used to identify the node and port affected by the Asynchronous Alert. Another possible implementation places this information into the Configuration Table, but the table defined below allows quick access to the information, especially if the TAG value of the ASYNC ALERT SMS is the entry number.

Table 14 - Master asynchronous alert table entry

Field	Description
MAA TAG	A two byte value consisting of the Asynchronous Alert TAG value
MAA PORT	A one byte value indicating the Master's port number used to send the ASYNC REPLY SMS.
MAA NODE IDENTIFIER	A pointer into the Configuration table to associate the Asynchronous Alert with a particular node.

9.2.3 Topology table

The Topology table shall be build and maintained by all Configutor nodes during the configuration process. The format of the Topology table is vendor specific, but at a minimum, the Topology table needs to have sufficient information to generate the Configuration table.

9.2.4 Configuration table

In the Configuration process (see 9.4) each Configutor node shall construct and maintain a Configuration table by walking the Web one node at a time. The table is essentially a list of all the nodes with pointers to represent the physical links.

---

[3]To obtain an Organizationally Unique Identifier (OUI) contact: Registration Authority Committee, The Institute of Electrical and Electronic Engineers, Inc., 445 Hoes Lane, Piscataway NJ 08855-1331, USA, (908) 562-3812.

The Master shall also issue a CONFIGURE PORT SMS to each port on every other node in the Web, including ports that are not currently Operational (see 11.2.3.).

The format of the Configuration table is vendor specific. One possible implementation contains the fields defined in Table 15.

Table 15 - Configuration table entry

Field	Description
UNIQUE ID	The Unique ID of the remote node.
TOTAL PORTS	The total number of ports implemented by the remote node. This indicates the length of Pointer array below.
MASTER PRIORITY	A 3-bit field indicating the priority of the remote node for becoming the Web Master. Zero indicates that the node is not capable of functioning as a Master (that is, that the node is a Responder node). Non-zero values are the node's priority for becoming Web Master, with 1 the lowest priority and 7 the highest.
PATH ADDRESS	The address of the primary path to the node. Together with local port this provides the information needed to address the remote node rapidly.
LOCAL PORT	The number of the Configutor node port, for the primary path to the remote node.
RETURN PATH ID	The RETURN PATH ID of the primary path to the node.
RETURN PATH	The return path address from the node to the Configutor node. This is a required field in most SMSs that the Configutor node sends to the node. Saving it in the Configuration table optimizes performance.
SMS OUTSTANDING	A flag that is set when the Configutor node has an outstanding CONFIGURE PORT, MASTER ALERT, QUERY PORT, QUERY PROTOCOL, QUERY SWITCH, or QUIESCE SMS. The flag is reset when the SMS is no longer outstanding (an appropriate response is received, including an associated ASYNC ALERT SMS).

A Configuration table entry optionally contains other information provided by the upper-level protocol (e.g., the node type and serial number).

Availability and performance are improved if a Web is configured with more than one path between two particular nodes. In this case the choice of the primary path and any alternative paths is vendor specific.

If a node has a port with an OPERATIONAL FLAG cleared then the Configutor nodes do not attempt to walk that link during Configuration process (this shall produce an ASYNC ALERT SMS with an ALERT CODE value of ADDRESSED PORT NOT OPERATIONAL.). If the port subsequently sets it's OPERATIONAL FLAG, each Configutor node receives a MASTER ALERT SMS with an ALERT CODE value of PORT NOW OPERATIONAL. Then each Configutor node shall walk the new link and adds nodes to its Configuration table until it encounters a single port node, another port with a cleared OPERATIONAL FLAG or it receives a Unique ID that is already known.

A Responder node shall not invoke the Configuration process or build a Configuration table.

9.2.5 Configutor table

In the configuration process each node shall construct a Configutor table from the information received in QUERY NODE SMSs. All nodes construct a Configutor table. The table is a list of the Unique ID of each Configutor node with its return path(s) and associated port(s). It is used during the processing of the QUIESCE SMS to clear Configutor node information, or by SMSs that use the RETURN PATH ID field to associate the RETURN PATH ID to a Return Path and port.

The format of the Configutor table is vendor specific. However each entry contains the fields defined in Table 16. The Configutor table optionally contains information needed by the upper level protocols.

Table 16 - Configutor table entry

Field	Description
UNIQUE ID	The Unique ID of a Configutor node.
NUMBER OF PATHS	The number of paths that the Configutor node is using to the node. This corresponds to the number of elements in Port array and return path array below.
PORT ARRAY	An array where each element is the node port for one return path to the Configutor node.
RETURN PATH ARRAY	An array whose elements correspond to the elements of Port array. Each element is a return path from the Responder node to the Configutor node.
RETURN PATH ID ARRAY	An array whose elements correspond to the elements of Port array. Each element is a value assigned by the node that the Configutor node is to use for any application SMS that is to use the corresponding return path in the return path array.

A node makes an entry in its Configutor table for each QUERY NODE SMS received, unless it specifies disable registration (DR bit set) or an equivalent entry already exists.

When a node executes an ordinary QUIESCE SMS it sets the NUMBER OF PATHS field in the Configutor table to zero and invalidates the associated return path id(s), but retains the UNIQUE ID entry along with any upper level protocol specific information. This quiesce action is performed when a previously active path from that Configutor node has been disconnected. The Configutor node sends the node a QUERY NODE SMS over each path that it intends to continue using. This ensures that any unused RETURN PATHs are removed from the Configutor table.

When a node executes a QUIESCE SMS with the RCTE bit set, it deletes the entire Configutor table entry for the specified UNIQUE ID. This action is performed when a Configutor node has been disconnected from the Web.

The Configutor table is cleared when a node executes a Total Reset or Absolute Reset Control frame. When one of the links to the node becomes Operational again each Configutor node receives an Asynchronous Alert and walks the link with QUERY NODE SMSs. The Master node issues a CONFIGURE PORT SMSs to each port. These actions rebuild the Configutor table.

> NOTE 7 - It is recommended that Responder nodes provide space for at least 64 entries in the Configutor table. If there is no space to make an entry when a node receives a QUERY NODE SMS then the node sets the ITF bit in the QUERY NODE REPLY SMS.

9.2.6 REFLECTION FLAG

The REFLECTION FLAG of a node, causes all ports on that node to either forward or reflect SAT and SAT' characters as defined in 8.5.2.

The REFLECTION FLAG is set if any of the following conditions occur:
  a)   the node is a single port node or switch node;
  b)   one of the ports of a dual port node becomes not Operational;
  c)   or a dual port node processes a CONFIGURE PORT SMS with the REFLECT bit set.

The REFLECTION FLAG is cleared when a dual port node with the OPERATIONAL FLAG set for both ports processes a CONFIGURE PORT SMS with the REFLECT bit cleared.

9.3 Beginning communication process

A port shall enter the Disabled state and Privileged mode after the following conditions occur:
  a)   POST;
  b)   a Total Reset Control frame is processed;
  c)   an Absolute Reset Control frame is processed;

    d)   or the Link ERP process exits with an ALERT CODE value.

The port shall use the following procedure to begin communication:

1) If the line fault detector indicates a line fault, then wait until there is no line fault. A line fault indicates that the remote node is not powered-on or that the cable is disconnected.

2) Transmit at least 200 DIS characters.

3) Put the port into the Enabled state.

4) Wait for the port to enter the Ready state, then set the port's OPERATIONAL FLAG.

5) Wait for 10 FLAG characters to be sent.

6) Ensure that buffer space is available to receive at least 1 frame of any type. Then send an RR character pair and clear the RR PENDING FLAG (see 8.4.3.)

7) Wait to receive an RR character pair from the remote port. Then clear the WAITING FOR RR FLAG.

Privileged frames may now be received and transmitted.

9.4 Configuration

Each Configutor node shall invoke a Configuration process to determine the other nodes that are present and their path address(es).

9.4.1 Configuration process

The Configuration Process shall involve the following steps:

1) Walk the Web (see 9.4.2) to build a Topology table (see 9.2.3).

2) Examine the Topology table to determine the Master. A Master shall be defined as the Configutor node with the largest value reported in the MASTER PRIORITY field. If more than one node has the same value for the MASTER PRIORITY field, then the node with the largest Unique ID shall be the Master.

3) Construct a Configuration table from the information in the Topology table.

4) Register with the nodes that it expects to perform I/O process with. If a node supports multiple protocols a QUERY PROTOCOL SMS shall be issued to determine the protocols supported.

5) If the node is to be the Master (see 12.1), then it shall internally places all of its operational ports in Normal mode.

6) The Master shall send a CONFIGURE PORT SMS for each port to the nodes in the Topology table. Operational ports are placed in Normal mode, and non-operational ports are left unchanged.

7) The Master shall send a MASTER ALERT SMS with an ALERT CODE value of BOTH PORTS OF LINK ARE IN NORMAL MODE for each link that has both ports in Normal mode. After configuring every port on a switch, the Master shall send a MASTER ALERT SMS with an ALERT CODE value of ALL OPERATIONAL PORTS OF ON THE SPECIFIED NODE ARE IN NORMAL MODE.

8) When all operational ports of the Web have been placed in Normal mode, the Master shall send a MASTER ALERT SMS with an ALERT CODE value of ALL OPERATIONAL PORTS ON ALL NODES IN THE WEB ARE IN NORMAL MODE.

9.4.2 Web Walking process

The Base node shall be set to the Configutor node performing the Web Walking process. All ports on the Configutor node shall be marked as unexplored.

The Web Walking process is defined in the following steps, and shall be performed until all ports are explored. The term Base node is defined as the node from which that the Master walks a string or loop.

1) The Configutor node selects a port on the base node and marks it as explored. The Web walking process is repeated for each unexplored operational port on the base node.

2) The Configutor node shall send a QUERY NODE SMS to the destination node incrementing the path address by one until a QUERY NODE REPLY is received reporting:

    i)   only one port operational indicating the end of a string has been reached;

    ii)   a Unique ID that already exists in the Topology table indicating a loop has been established;

iii)   or the TOTAL OTHER PORTS field value exceeds two indicating that a switch has been discovered.

3) The Configutor node may send a QUERY PROTOCOL SMS and QUERY PORT SMSs as desired.

4) If a switch is discovered the Configutor node shall send a QUERY SWITCH SMS to determine the number of operational ports on the switch. The base node shall be set to that switch node and the Web Walking process shall be recursively called. The port indicated in the QUERY NODE REPLY SMS shall be marked as explored.

9.5 Resets

This sub clause defines the various types of reset that are associated with SSA-TL1.

9.5.1 Link reset

The Link Reset Control frame is used during the Link ERP process to recover from a link error(see 10.1).

The Link Reset frame is confined to a single link. It shall never forwarded by a router from one link to another. Therefore, the STATUS field contains the Link Status Byte that is used by the Link ERP process (see 10.1.2).

Every port shall be able to originate and invoke a Link Reset frame. A Link Reset frame shall only be originated by a port in the Check state, regardless of the WAITING FOR RR flag state.

Link Reset frames shall be acknowledged but are not subject to pacing or sequence numbers. The transmitter shall not inspect or change the WAITING FOR RR FLAG or the TRANSMIT SEQUENCE NUMBER. The receiver shall not inspect or change the RR PENDING FLAG or the RECEIVE SEQUENCE NUMBER

The receiver shall recognize a Link Reset whenever it is in the Ready or Check states, regardless of whether its inbound buffers are full. If the frame is valid the port shall return an ACK character pair and enter the Check state if it has not already done so (the ACK character pair is useful for fault isolation since it indicates that the link is Operational even if the node processor has hung). The contents of the frame buffers shall not be changed by the receipt of a Link Reset since they are needed by the Link ERP process. The port mode is not affected unless the Link ERP process exits with an ALERT CODE value.

9.5.2 Total reset or absolute reset

Absolute Reset shall invoke at least the same functions as Total Reset. Optionally, some implementations treat Absolute Reset as a stronger form of Total Reset, invoke additional internal resets and invoke the POST. If no additional resets are done, the destination node processes an Absolute Reset frame the same as a Total Reset frame. Storage requirements for handling Absolute Reset frames may be shared with Total Reset frames.

A Configutor node may reset a destination node by originating either a Total Reset or an Absolute Reset Control frame that is addressed to the node. All Configutor nodes shall be capable of originating Total Reset, Reserved or Absolute Reset Control frames. The destination node shall process a Total Reset Control frame in the order as follows:

a) Perform optional resets and optional POST.

b) Clear the OPERATIONAL FLAG.

c) All ports are set to the Disabled state and Privileged mode.

d) Other node functions are optionally re-initialized (e.g., the interface processor in a node) by an Absolute Reset frame.

e) The frame buffers shall be initialized.

f) The TRANSMIT POINTER and RETRY POINTER shall be cleared.

g) The Configuration table in a Configutor node shall be cleared.

h) The Configutor table shall be cleared.

i) The Port table shall be cleared.

j) The Queued Asynchronous Alert table shall be cleared.

k) The Asynchronous Alert Address table shall be cleared.

l) Any pending Asynchronous Alerts shall be cleared.

m) The node shall initialize the following items for each port (see 11.2.3):

1) A QUOTA = 1;
2) B QUOTA = 4;
3) REFLECT = 0b;
4) EUDC = 0b;
5) LINK ERP COUNTER = 0;
6) and ALARM THRESHOLD = 10.

n) The Beginning Communication process shall be invoked.

When a Configutor node receives a Total Reset or Absolute Reset frames , it shall start a 10 second time-out and shall wait to receive a MASTER ALERT SMS with an ALERT CODE value of RECONFIGURATION REQUIRED. The Configutor node shall wait to configure the Web until after the time-out expires or the MASTER ALERT SMS is received. This is different from when a Configutor node shall configure the Web immediately following a power-on reset.

The Total Reset or Absolute Reset frames shall not be subject to pacing, acknowledgments or sequence numbers. The transmitter shall sends the Total or Absolute Reset frame immediately after it finishes the current frame, if any. The receiver does not inspect or change the RR PENDING FLAG, ACK PENDING FLAG or RECEIVER SEQUENCE NUMBER.

A transmitter shall send a minimum of 200 FLAG characters prior to originating a Total Reset or Absolute Reset frame while in the Enabled State to allow the remote port time to synchronize if necessary. A transmitter shall be able to send the Total Reset or Absolute Reset frames when the port is in the Enabled or Ready state. The receiver shall recognize the Total Reset or Absolute Reset when the port has achieved character synchronization in any state, regardless of whether its inbound frame buffers are full.

The router in dual port or switch nodes shall be able to store and forward at least one Total Reset, Reserved or Absolute Reset Control frame. Since Control frames are not paced, dedicated storage is required if the frame buffers for Privileged, Reserved and Application frames are occupied. If another Total Reset, Reserved or Absolute Reset Control frame is received while the dedicated storage is in use then the node may discard it. If the output port is in the Disabled or Check state the Total Reset or Absolute Reset frame shall be held until it enters the Enabled or Ready state. Consequently a Total Reset or Absolute Reset frame shall not be sent during the Link ERP process. The router shall be able to forward a Total Reset or Absolute Reset frame even when the output port's OPERATIONAL FLAG is cleared.

9.5.3 Power on reset

Power on Reset is an internal function that resets the entire node at power-on, after a catastrophic internal error or when a node is disconnected from all other nodes.

When a node invokes a Power-on reset, it shall initialize the SSA-TL1 functions in the order as follows:

a) All ports shall enter Wrap mode.
b) The node shall invoke an Absolute Reset.

9.5.4 Web Reset process

The Web Reset process may be used by the Master when communication over the Web is blocked and the cause cannot be identified.

The Master prepares to invoke a Web Reset process in the order as follows:

a) The Master shall clear its Configuration table and shall make an entry for itself.
b) The Master shall place each of its ports in the Disabled state and shall clear its OPERATIONAL FLAG. The Master shall execute the Beginning Communication process (see 9.3) for each of its ports.

The Master shall walk the entire Web resetting each node except itself. To avoid ambiguity, the following terminology is used:

a) Current node is the node that is about to be reset.
b) Previous port is the port nearer to the Master on the final link in the selected path from the Master to the Current node.

c) Base node is the node from which the Master walks a string or loop.

For Webs that include switches, the Master walks one level of switching at a time. Initially, the Master is the Base node for resetting all nodes that are accessible without going through a switch node. Then, the Master uses each switch it discovers as the Base node for resetting all nodes that are accessible via a single port on each switch, and so on.

Initially, the Master marks all ports on each Base node as unselected. A Base node port becomes Selected when the Master uses it as the origin for walking a string or loop or when the Master encounters it as the last node of a string or loop.

The process to reset a loop or all of the strings connected to a particular Base node is in the order as follows:

a) The Master selects an Unselected port on the Base node as the new Previous port and marks it as selected.

b) The Master issues a CONFIGURE PORT SMS (or the internal equivalent) to Previous port. This ensures that the Master receives subsequent Asynchronous Alerts from the Previous port.

c) The Master sends a QUERY NODE SMS to the Current node and starts a 5 ms timer.

d) If the Master does not receive a QUERY NODE REPLY SMS before the timer expires, it continues with step 5. Otherwise, if the Master receives a QUERY NODE REPLY SMS, the Master shall compare the UNIQUE ID field in the QUERY NODE REPLY SMS with each entry in its Configuration table. If the Unique ID is in the Configuration table then the Master shall mark the port indicated by the Current port field in QUERY NODE REPLY SMS as Selected and shall skip to step 9; the Master has returned on a loop or a cyclic path to itself or a switch that it has already reset. If the Unique ID is not in the Configuration table, the Master shall continue with step 5; since the Current node has not yet been reset.

e) The Master sends a Total Reset to the Current node, starts a 5 ms timer, and waits to receive an ASYNC ALERT SMS with an ALERT CODE value of PORT NOW OPERATIONAL from Previous port. Except for issuing an ASYNC REPLY SMS, the Master shall ignore any other ASYNC ALERT SMSs (for example the Total Reset may cause the Previous port to initially invoke the Asynchronous Alert process with an ALERT CODE value of REMOTE PORT DISABLED).

f) If the Master does not receive an ASYNC ALERT SMS with an ALERT CODE value of PORT NOW OPERATIONAL from Previous port before the 5 ms timer expires then the Master shall skip to step 9; the Current node shall be assumed to be not operational. Otherwise the Master shall send a QUERY NODE SMS to Current node and starts a 5 ms timer.

g) If the Master does not receive a QUERY NODE REPLY SMS before the timer expires, then the Master shall skip to step 9; the Current node shall be assumed to be not operational. Otherwise the Master shall make an entry in its Configuration table for the Current node.

h) If the Current node is a single port node, the Master shall continue with step 9. If the Current node is a dual port node, the Master updates the Previous port to refer to the port on Current node that is further from the Master and returns to step 2. If the Current node is a switch, the Master shall mark the port indicated by the PORT field in QUERY NODE REPLY SMS as Selected and shall continue with step 9.

i) If there is an Unselected port remaining on the Base node, the Master returns to step 1. Otherwise the walk from the current Base node is complete and the Master selects a new Base node if one exists.

When the Web Reset process is complete the Master shall clear its Configuration table and shall invoke the Configuration process described in 9.4. The Master shall also send a MASTER ALERT SMS with an ALERT CODE value of RECONFIGURATION REQUIRED to each other Configutor node. This shall cause each other Configutor node to clear its Configuration table and invoke the Configuration process.

10 Error handling

This clause defines the strategy for recovering errors in a complex SSA Web with multiple Configutor nodes. The strategy ensures data integrity and it minimizes the impact to other operations. For example, an error does not result in bad data being written to a device. An error affects only the commands and nodes currently using the failing link or node.

10.1 Link ERP

SSA-TL1 defines a Link Error Recovery Procedure (Link ERP) to recover link errors at the frame level. This has the following benefits:

    a)   The upper-level protocol is simplified since frame error recovery is transparent if it is successful.

    b)   There is normally no need to terminate any commands when a transient error occurs

    c)   There is no uncertainty about the state of the remote node.

If the Link ERP process determines that a transmission error occurred then it attempts to recover from the error itself. If recovery is successful the Link ERP process terminates and the upper-level protocol continues unaware of the error.

The port that detects the Link Error invokes the Link ERP process. During the Link ERP process, a Link Reset frame shall be sent to the remote node, that shall trigger the remote node to invoke the Link ERP process (if it did not also detect an error and invoke it's own Link ERP process). The Link ERP process is coordinated between the two ports so that they remain synchronized.

The Link ERP process cannot recover from some errors transparently (e.g., hardware errors or permanent line faults). The Link ERP process has been designed so that both nodes always recognize an unrecoverable error and remain synchronized. In these cases the Link ERP exits and attempts to alert the Master via the Asynchronous Alert process (see 10.3).

The basic principles of the Link ERP process are as follows:

    a)   Only the failing link invokes the Link ERP process. Other links in the Web are not involved.

    b)   The Link ERP process does not recover Control frames.

    c)   In normal operation the transmitter does not discard a Privileged, Application, or Reserved frame until it has received an ACK character pair. This indicates that the frame has been received correctly by the remote port. Therefore when an error occurs the affected frame(s) are still available for re-transmission without reference to the upper-level protocol.

    d)   When an error is detected both ports enter the Check state, invoke the Link ERP process and exchange status by means of Link Resets.

    e)   Recovery is invoked separately for each line. Each port is responsible for recovering frames that were lost on its outbound line. If the transmitter starts sending another frame before it receives an ACK character pair and a recoverable error occurs, 1 or 2 frames are retransmitted.

    f)   Before restarting communication, the Link ERP process forces the port into the Disabled state. This synchronizes the Link ERP process in both nodes and allows an orderly restart with the same mechanisms that are used at power-on.

    g)   The link protocol and ERP are designed to eliminate the loss or duplication of frames when an error occurs.

10.1.1 Link Errors

Except where explicitly stated the following errors are only detected when a port is in the Ready state before the error. Errors are ignored if a port is not in the Ready state.

When an error is detected, the port enters the Check state and interrupts the node processor. No further Privileged, Reserved or Application frames are accepted or acknowledged until the node returns to the Ready state.

10.1.1.1 Hardware error

The Hardware error indicates when a port detects an internal hardware error (e.g., a parity check).

10.1.1.2 Line fault

The Line Fault error indicates when the output of the line fault detector is active and the port is not in the Disabled state or Wrap mode (e.g., the cable is open, short circuited or the remote node is powered off).

10.1.1.3 ACK time-out

The ACK Time-out error indicates that the source port has not received an ACK character pair within the specified time after transmitting the trailing FLAG character of a frame other than a Total Reset, Reserved or Absolute Reset Control frame.

10.1.1.4 LOSS OF SYNCHRONIZATION ERROR

The LOSS OF SYNCHRONIZATION ERROR indicates that the clock recovery circuits in the receiver detected a synchronization error in the Ready state.

10.1.1.5 CODE VIOLATION ERROR

The CODE VIOLATION ERROR indicates that the receiver has not detected a LOSS OF SYNCHRONIZATION ERROR, and has either decoded a character that is not a data or Special character or has decoded a character that causes a disparity error.

10.1.1.6 PROTOCOL ERROR

The PROTOCOL ERROR indicates the receiver has neither detected a LOSS OF SYNCHRONIZATION ERROR nor has detected a CODE VIOLATION ERROR, and a port receives an incorrect sequence of valid characters as follows:

a) a frame with less than 6 data characters between 2 FLAG characters (e. g., noise corrupting or manufacturing a FLAG character);

   NOTE 8 - It is recommended that a short, and therefore invalid, Privileged, Reserved or Application frame with less than 7 data characters between 2 FLAG characters also be identified as a PROTOCOL ERROR.

b) a Privileged, Reserved or Application frame and no buffer is available, (i.e., when the RR PENDING FLAG is set);

c) an unexpected RR character pair (i.e., when the WAITING FOR RR FLAG is cleared);

d) an isolated RR character. One half of a link hangs if an RR character pair is lost without any errors being detected, (e.g., if both RR characters are changed to FLAG characters while the link is idle). This is extremely unlikely but the Healthy Web process shall detect and correct this case;

e) an unexpected ACK character pair (i.e., when the WAITING FOR ACK FLAG is cleared);

f) an isolated ACK character. If an ACK character pair is corrupted then the transmitter also detects an ACK Time-out error;

g) a NUL character with no intervening data character since the last FLAG character:

h) an ABORT character with no intervening data character since the last FLAG character:

i) an ABORT character that is not immediately followed by a FLAG character;

j) an isolated DIS character.

10.1.1.7 CRC ERROR

The CRC ERROR indicates that a frame was received that was not aborted and none of the receiver errors with a lower value than CRC ERROR has occurred (see Table 18).

10.1.1.8 SEQUENCE ERROR

The SEQUENCE ERROR indicates that a received frame has FSN ≠ RSN, the frame has not been aborted, the frame is not a Control frame and none of the receiver errors has occurred with a lower value than SEQUENCE ERROR (see Table 18).

10.1.1.9 FRAME REJECT ERROR

The FRAME REJECT ERROR indicates that a frame is received correctly with none of the receiver errors above, the frame has not been aborted, and the frame is unacceptable for one of the following reasons.

a) The frame contains more than 139 data characters. The receiver shall continue to accumulate the CRC until the trailing FLAG, since a CRC ERROR Link ERP exit takes precedence over a FRAME REJECT Link ERP exit.

b) The destination port determines that the CONTROL field contains a Reserved FRAME TYPE or Reserved RESET TYPE field (see 7.1).

c) The first byte of the Path component is invalid (see 8.2.1, 8.2.2 and 8.2.3).

d) The destination port determines that the Channel address is invalid or the destination port cannot accept a frame on the specified Channel (see 7.4.4).

e) The length of the DATA field in a Control frame is not zero.

f) The destination port determines that the length of the DATA field in an SMS frame exceeds 32 bytes.

g) The destination port determines that the length of the DATA field in a frame exceeds 128 bytes.

h) The length of the DATA field in a Data frame is not an integer multiple of 8 bytes, the frame does not contain the last byte of the transfer and the destination port cannot accept it.

10.1.2 Link status byte

During error recovery the Link ERP process in each node builds a Link Status Byte as defined in Table 17 and sends it to the other node in the STATUS field of a Link Reset frame.

Table 17 - Link status byte

Bit 7	6	5	4	3	2	1	Bit 0
HW	LF	ACK	RECEIVER ERRORS			RSN	

The HW bit when set indicates that the port detected an internal hardware error.

The LF bit when set indicates that the port detected a line fault. It is provided for diagnostic information only and it is not referenced by the Link ERP process in the remote node.

The ACK bit when set indicates that the transmitter timed-out while waiting for an ACK character pair. It is provided for diagnostic information only and it is not referenced by the Link ERP process in the remote node.

The RECEIVER ERRORS field is defined in Table 18.

Table 18 - Receiver errors

Value	Description
000b	NO ERROR
001b	LOSS OF SYNCHRONIZATION ERROR
010b	CODE VIOLATION ERROR
011b	PROTOCOL ERROR
100b	CRC ERROR
101b	SEQUENCE ERROR
110b	FRAME REJECT ERROR
111b	reserved

When two or more errors occur simultaneously the lowest RECEIVER ERROR field value is reported (i.e. SEQUENCE ERROR is lower priority than CRC ERROR).

The RSN field specifies the RECEIVE SEQUENCE NUMBER for the next Privileged, Reserved or Application frame that was acknowledged by the port. It is needed by the Link ERP process in the remote node.

10.1.3 Link ERP process definition

To facilitate cross-referencing, the Link ERP exits are identified by an ALERT CODE value that corresponds to the ALERT CODE field used by the Asynchronous Alert process.

The Link ERP process shall not be invoked from within itself (i.e. it is non-recursive). The implementation shall protect against the Link ERP process looping if there is a permanent error.

> NOTE 9 - The following is an example of one method that is used to protect against the Link ERP process from looping. Each invocation of the Link ERP process increments a RETRY COUNTER that is reset to zero periodically by a timer. If the number of retries in one period of the timer exceeds a maximum value then the Link ERP process waits 25 ms to ensure that the remote port recognizes an unrecoverable error, it places the port in the Disabled state, and it exits with an ALERT CODE value of RETRY LIMIT EXCEEDED. This example also protects against excessive use of the ERP in the event of severe external noise.

The port that detects the error enters the Check state and invokes its Link ERP process. Any Link ERP exit shall terminate the Link ERP process and invoke the Asynchronous Alert process as defined in 10.3. The Link ERP process is defined in the order as follows:

a)   The Link Status Byte is built. If the port detects a line fault that persists for more than 1 ms the port enters the Disabled state, and Link ERP process exits with an ALERT CODE value of PERMANENT LINE FAULT.

b)   If the port is not receiving characters, the port enters the Disabled state, and the Link ERP process exits with an ALERT CODE value of NO CHARACTERS RECEIVED.

c)   If the port is detecting DIS characters the port enters the Disabled state and the Link ERP process exits with an ALERT CODE value of REMOTE PORT DISABLED.

d)   A Link Reset frame is transmitted. If no acknowledgment is received the Link Reset frame is retransmitted once. If no acknowledgment is received to the retransmitted Link Reset frame, then the Link ERP process waits 25 ms to ensure that the remote port recognizes an unrecoverable error, sets the port to the Disabled state, and exits with an ALERT CODE value of LINK RESET FAILED.

e)   If the LINK RESET RECEIVED FLAG is cleared and a Link Reset frame has not been received within 5 ms, the Link ERP process waits 25 ms to ensure that the remote port recognizes an unrecoverable error, places the port in the Disabled state, and exits with an ALERT CODE value of LINK RESET FAILED.

f)   If a hardware error is detected, the Link ERP process sets the port to the Disabled state and exits with an ALERT CODE value of HARDWARE ERROR.

g)   If FRAME REJECT ERROR is indicated by the port's Link Status Byte, the Link ERP process sets the port to the Disabled state, and exits with an ALERT CODE value of FRAME REJECT.

h)   The number of outbound frames with an acknowledgment outstanding is calculated. When using strict cyclic buffers this is obtained as follows:

$$Q = (\text{TRANSMIT POINTER} - \text{RETRY POINTER}) \bmod N$$

where N is the number of transmit buffers provided. Q is 0 or 1 frames. The number of outbound frames that the local port is expecting an acknowledgment for but have not been received by the remote port is also calculated, where local refers to the ports TRANSMIT SEQUENCE NUMBER and remote refers to the RSN field received in the Link Status Byte.

$$P = (\text{local TRANSMIT SEQUENCE NUMBER} - \text{remote RSN}) \bmod 4$$

where P is less than or equal to Q. If either Q > 1 or P > Q then the Link ERP process waits 25 ms to ensure that the remote port recognizes an unrecoverable error, sets the port to the Disabled state and it exits with an ALERT CODE value of INVALID RETRY STATUS.

i)   The Link ERP process arranges to resend the number of frames indicated by subtracting P from its TRANSMIT POINTER, modulo N. Those outbound buffers that do not need to be retransmitted shall be discarded using the following algorithm.

```
Do while RETRY POINTER ≠ TRANSMIT POINTER;
        Discard buffer at RETRY POINTER;
        Increment RETRY POINTER modulo N;
    End;
```

The frame buffers and TRANSMIT POINTER are set to retransmit the proper number of frames after the Link ERP re-establishes communication following the port being set to the Disabled state.

j)   If the port has received a frame containing any of the errors listed in Table 18, then the appropriate inbound buffer contents shall be discarded. The ERP places the port in the Disabled state. The ERP waits until the receiver detects DIS characters. This is required to synchronize the local and remote ports Link ERP process's and prevent the transmitter sending RR while the remote port is still in the

Check state.
If the receiver does not detect DIS characters within 5 ms after the local port is disabled then the ERP exits with an ALERT CODE value of TIME-OUT WAITING FOR DISABLED STATE.

k) The port is set to the Enabled state.
If the receiver detects a FLAG character within 5 ms of placing the port in the Enabled state, the port is set to the Ready state, otherwise the Link ERP process exits with an meaning code of TIME-OUT WAITING FOR READY STATE.

l) The ERP terminates successfully.

10.1.3.1 Timing restrictions

To avoid false time-outs the implementation shall ensure that the following paths in the Link ERP process do not exceed 5 ms each:

a) from receiving a Link Reset frame to sending a Link Reset frame;

b) from the later of sending or receiving a Link Reset frame to placing the local port in the Disabled state;

c) from the later of detecting DIS characters or placing the local port in the Disabled state to placing the local port in the Enabled state.

10.1.3.2 ERP exits

If the Link ERP process exits unsuccessfully then each node shall take the following actions in the order shown:

a) Clear the port's OPERATIONAL FLAG.

b) Set the port to Privileged mode. The transmitter discards Application frames. This avoids frames backing up and blocking other traffic. It also prevents any further data from being transferred through the affected port.

c) Invoke the Asynchronous Alert process (see 10.3) with the ALERT CODE specified in the Link ERP exit.

d) Restart the affected port, as described in 9.3.

10.1.3.3 Example

The operation of the Link ERP process is illustrated in Table 19. Port 1 sends 2 frames back-to-back. Port 2 receives the first frame correctly but the ACK character pair is corrupted by noise. Port 1 then detects a code violation and aborts the second frame. After the Link ERP process completes Port 1 retransmits the second frame. To illustrate the operation of the TRANSMIT POINTER (TP) and the RETRY POINTER (RP) it is assumed that Port 1 has at least 3 transmit buffers.

Table 19 - Recovery from corrupted ACK character pair

Port 1	Line →	← Line	Port 2
TP=0, RP=0, TSN=0, RSN=0			TP=0, RP=0, TSN=0, RSN=0
Tx frame 1	Frame 1		
	"	RR character pair	Tx RR character pair
	"		
	"		
TSN=1, TP=1			Good CRC
Tx frame 2	Frame 2		RSN=1
	"	** Error **	Tx ACK character pair
Code violation	"	RR character pair	Tx RR character pair
Enter Check state	"		
Invoke Link ERP process	ABORT		RSN=1 (still)
Wait for Tx complete			
Assemble LSB			
Tx Link Reset	Link reset		
Wait to Rx Link Reset		ACK character pair	Tx ACK character pair
"			Enter Check state
"			Invoke Link ERP process
"			Wait for Tx complete
"			Assemble LSB
"		Link reset	Tx Link Reset
Tx ACK character pair	ACK character pair		Q=0, P=0
Q=1, P=0			
Discard 1 frame			
RP =1			
Enter Disabled state	DIS	DIS	Enter Disabled state
TSN=0, RSN=0	"	"	TNS=0, RSN=0
Wait to detect DIS	"	"	Wait to detect DIS
Enter Enabled state	FLAG	FLAG	Enter Enabled state
Wait to Rx FLAG			Wait to Rx FLAG
Enter Ready state			Enter Ready state
Tx frame 2	Frame 2		
	"	RR character pair	Tx RR character pair
	"		
	"		Good CRC
TSN=1, TP=2			RSN=1
		ACK character pair	Tx ACK character pair
RP=2			

10.2 Invalid SMS field handling

This clause discusses invalid field handling of data fields within frames. It is assumed that the frame has been successfully received by the hardware. This discussion centers on the processing of the DATA field (i.e. the contents of the SMS).

The order of processing is critical and matches the following order.

10.2.1 Unknown SMS code

An SMS received with an unknown SMS CODE cannot be parsed since it cannot be determined if the SMS has either a RETURN PATH or RETURN PATH ID field, and therefore shall invoke an Asynchronous Alert process with an ALERT CODE value of UNKNOWN SMS.

10.2.2 Unsupported SMS code

An SMS received with a known SMS CODE that is either not supported by the node, or has the incorrect FRAME TYPE shall invoke an Asynchronous Alert process with an ALERT CODE value of UNKNOWN SMS.

10.2.3 SMS too small

If the SMS length is shorter than specified under clause 11, an Asynchronous Alert process shall be invoked with an ALERT CODE value of SMS TOO SHORT.

It is not an error to receive an SMS longer than the specified length.

10.2.4 Invalid RETURN PATH ID

An Invalid RETURN PATH ID is a RETURN PATH ID not contained in the Configutor table. No RETURN PATH and port is associated with it and therefore, no response is sent to the specified Configutor node.

If the Responder node receives an SMS with an invalid RETURN PATH ID, an Asynchronous Alert process is invoked with an ALERT CODE value of INVALID RETURN PATH OR RETURN PATH ID. When the Master generates the MASTER ALERT SMS, this causes all Configutor nodes to issue a QUERY NODE SMS to the associated node to rebuild its Configutor table.

10.2.5 Invalid tag

An SMS sent to a Configutor node with an invalid TAG, or a valid TAG with an improper response causes the Configutor node to ignore the SMS; and optionally log the error. In the case of data transfer SMSs with an invalid TAG, the protocol layer issues the appropriate upper level protocol response to abort the I/O process if possible (i.e., ABORT TAG SMS for SSA-S2P or SSA-S3P).

10.2.6 Invalid context

When an SMS is received out of context, it shall be discarded and the event shall be logged by a Configutor node or ignored by a Responder node. Examples of out of context SMSs include the following.

   a) DATA REPLY SMS when no associated DATA READY SMS was issued.

   b) DATA REQUEST SMS was received for an I/O process that did not need to write data.

   c) DATA READY SMS was received for an I/O process that did not need to read data.

   d) ASYNC REPLY SMS with a valid TAG value without an outstanding ASYNC ALERT SMS.

   e) QUERY PROTOCOL REPLY SMS with a valid TAG value not associated with an outstanding QUERY PROTOCOL SMS.

   f) QUERY PORT REPLY SMS with a valid TAG value not associated with an outstanding QUERY PORT SMS.

10.2.7 Non-zero reserved field or reserved code value

An otherwise valid SMS with a non-zero reserved field or a reserved code value is parsed and the RETURN PATH ID or RETURN PATH field is used to send the appropriate response (excluding Unknown or Invalid SMS codes). The response for each transport layer SMS is defined in Table 20.

Table 20 - Response to non-zero reserved fields or reserved code values

SMS Name	Resulting SMS	Notes
QUERY NODE	Normal QUERY NODE REPLY	Ignore reserved field/values
QUERY NODE REPLY	None	Ignore reserved field/values
CONFIGURE PORT	RESPONSE SMS with RETURN CODE of INVALID FIELD	
RESPONSE	Discard	System dependent retry
ASYNC ALERT	ASYNC REPLY SMS with RETURN CODE of INVALID FIELD	Generate MASTER ALERT
MASTER ALERT	RESPONSE SMS with RETURN CODE of INVALID FIELD	
QUIESCE	RESPONSE SMS with RETURN CODE of INVALID FIELD	
ASYNC REPLY	None	No resend needed, clear the ASYNC ALERT data.
QUERY PROTOCOL	RESPONSE SMS with RETURN CODE of INVALID FIELD	
QUERY PROTOCOL REPLY	Discard	System dependent retry
QUERY PORT	RESPONSE SMS with RETURN CODE of INVALID FIELD	
QUERY PORT REPLY	Discard	System dependent retry
DATA READY	Discard	Cancel I/O Process
DATA REPLY	RESPONSE SMS with RETURN CODE of INVALID FIELD	
DATA REQUEST	Discard	Cancel I/O Process

10.3 Asynchronous alert process

The Asynchronous Alert process is given an ALERT CODE value, a port, and a frame (if applicable). All tests are based on the various tables associated with the Port parameter. The node then invokes one the following options.

   a) If a QUERY NODE REPLY SMS has not sent since the last power on, Total Reset or Absolute Reset then discard the data and exit the Asynchronous Alert process.

   b) Otherwise if the AA VALID flag is cleared, or the Queued Asynchronous Alert table is not empty, then queue the parameters in the Queued Asynchronous Alert table and exit the Asynchronous Alert process.

   c) Otherwise, queue the Asynchronous Alert data and invoke the in the De-queue Asynchronous Alert process with the Port parameter (see 10.4).

The receipt of the ASYNC ALERT SMS by the Master invokes the ASYNC ALERT SMS Handling process

10.4 De-queue asynchronous alert process

The De-queue Asynchronous Alert process is invoked by the Asynchronous Alert process for the first Asynchronous Alert following a valid CONFIGURE PORT SMS (AA VALID flag set in the Port table) or by the receipt of an ASYNC REPLY SMS that terminates an outstanding ASYNC ALERT SMS.

The De-queue Asynchronous Alert process is given a Port parameter. All tests are based on the various tables associated with the Port parameter. The De-queue Asynchronous Alert process performs the following steps in the order shown:

   a) If the Queued Asynchronous Alert table is empty then exit the De-queue Asynchronous Alert process

   b) Build an ASYNC ALERT SMS using the port (PORT field), ALERT CODE value, and frame (CONTROL, CHANNEL, and FRAME DATA fields). The TAG field value and path for the ASYNC ALERT SMS are obtained from the AA TAG and AA RETURN path fields of the Port table respectively.

   c) Send the ASYNC ALERT SMS. If no ASYNC REPLY SMS with a matching TAG and ALERT CODE value is received within 500 ms, then resend the ASYNC ALERT SMS. Repeat this step indefinitely until either a matching ASYNC REPLY SMS is received, or the port is reset by a Power On, Total, or Absolute Reset.

10.5 ASYNC ALERT SMS handling process

Asynchronous Alerts are used to report events that are outside the scope of the Link ERP process.

   a) A new link is connected to the Web.

   b) A router receives a frame that is addressed to a port that is not Operational.

   c)   A node receives an invalid or unknown SMS.

   d)   A node detects a protocol error.

   e)   A node detects a critical environmental error.

   f)   A node detects a non-critical warning error.

   g)   A Link ERP process exit.

   h)   A node detects a recovered Link ERP process threshold is exceeded.

Upon receipt of an ASYNC ALERT SMS, the Master optionally logs the error. Any other action required to handle the different ALERT TYPE field values is defined in the following sub clauses.

10.5.1 General ASYNC ALERT SMS handling process

The receipt of an ASYNC ALERT SMS with the following TYPE field values shall invoke the general ASYNC ALERT SMS Handling process.

   a)   ADDRESS ERROR is used when a frame is addressed to a port with a cleared OPERATIONAL flag. The port detecting the error remains in Normal mode.

   b)   UNKNOWN SMS is used when the SMS is not recognized by the node.

   c)   INVALID SMS is used when the SMS is deemed invalid.

   d)   PROTOCOL ERROR is used when an SMS is received out of order.

   e)   ENVIRONMENTAL ERROR is used when a critical system failure is detected. The handling of the error is system dependent, but often results in the suspension of all activity in the system.

   f)   ERROR THRESHOLDS is used when a non-critical system degradation is detected by exceeding a predetermined number of recovered link errors. The handling of this error is system dependent, but often results in system activity continuing (with a possible rerouting of future activity) with the possible generation of a repair action.

The general ASYNC ALERT SMS Handling process is defined in the order as follows:

   a)   Build an ASYNC REPLY SMS using the TAG and ALERT CODE values from the ASYNC ALERT SMS.

   b)   Send the ASYNC REPLY SMS using the ADDRESS field in the ASYNC ALERT Address table entry associated with the TAG field value of the ASYNC ALERT SMS.

   c)   Build a MASTER ALERT SMS using the port (PORT field), ALERT CODE value, and frame (CONTROL, CHANNEL, and FRAME DATA fields).

   d)   Send the MASTER ALERT SMS to all Configutor nodes using the PATH ADDRESS field of the Configuration table entries.

   e)   Log the error.

   f)   Wait for a RESPONSE SMS from each of the Configutor nodes for the MASTER ALERT SMS. If any Configutor node fails to return a RESPONSE SMS associated with the MASTER ALERT SMS within 1 second, the Web Reset process shall be invoked (see 9.5.4).

The receipt of the MASTER ALERT SMS invokes the Master Alert Handling process by the Configutor node.

10.5.2 PORT NOW OPERATIONAL type specific handling process

When a new link is connected to the Web a port on one of the existing nodes sets it's OPERATIONAL FLAG and sends an ASYNC ALERT SMS to the Master with an ALERT CODE of PORT NOW OPERATIONAL. The PORT NOW OPERATIONAL Type specific handling is performed in the order shown:

   a)   Perform the general ASYNC ALERT SMS Handling process (see 10.5.1).

   b)   The Master shall walk the new link by issuing QUERY NODE SMSs to the new node(s). Each node shall return a QUERY NODE REPLY SMS.

   c)   The Master shall issue a CONFIGURE PORT SMS specifying SET NORMAL MODE to those ports that are Operational on the new nodes that it has walked. Each port shall return a RESPONSE SMS.

   d)   The Master shall issue a Master alert to each other Configutor node with an ALERT CODE value of BOTH PORTS OF LINK ARE IN NORMAL MODE for each new link whose ports are now in Normal mode. If the Web is now a loop, the Master optionally issues a single MASTER ALERT SMS with an ALERT CODE

value of ALL OPERATIONAL PORTS ON ALL NODES IN THE WEB ARE IN NORMAL MODE in place of the individual Master alerts with an ALERT CODE value of BOTH PORTS OF LINK ARE IN NORMAL MODE.

e) After the other Configutor nodes have returned a RESPONSE SMS for each Master alert, no additional ASYNC ALERT SMSs with a error code of PORT NOW OPERATIONAL are processed until this procedure is completed.

10.5.3 WARNING type specific handling process

The actions taken when a WARNING TYPE FIELD value are received are separated into two groups, the MASTER PRIORITY CHANGED alert code value and all others.

The WARNING Type MASTER PRIORITY CHANGED ALERT CODE Asynchronous Alert is generated when the Master priority level of a Configutor node changes. The actions taken are defined in 12.1.

The WARNING Type excluding MASTER PRIORITY CHANGED ALERT CODE Asynchronous Alert type is generated when a non-critical system degradation is detected. The handling of this error is system dependent, but often results in system activity continuing with the generation of a repair action. The Master alerts all Configutor nodes who takes the appropriate system dependent action.

10.5.4 LINK ERP EXITS type specific handling process

An ERP exit indicates that an unrecoverable link error has occurred. Initially the Master is informed by one or two ASYNC ALERT SMSs from the nodes that are connected by the failing link. The Master returns an ASYNC REPLY SMS and recovery proceeds as follows. The LINK ERP EXITS Type specific handling is performed in the order shown:

a) Perform the general ASYNC ALERT SMS Handling process (see 10.5.1).

b) If the ASYNC ALERT SMS has an ALERT CODE value of either LINK RESET FAILED or TIME-OUT WAITING FOR READY STATE following an ERP exit then the Master sends a Total Reset frame to the failing node.

c) The Master invalidates the entries in the Configuration table for the affected link(s) and node(s). If the failing node recovers after the Total Reset then the remote node(s) invoke the Asynchronous Alert process with an ALERT CODE value of PORT NOW OPERATIONAL. This causes the affected links and nodes to be reconfigured.

d) If the ASYNC ALERT SMS indicates any Link ERP process exit, the Master shall invalidate the entries in the Configuration table for the affected link(s).

e) If the Master determines that a previous Configutor node has been removed from the Web by the error then it sends a QUIESCE SMS with the RCTE bit set to all remaining nodes on behalf of the missing Configutor node.

f) If the link is still configured the Master sends a CONFIGURE PORT SMS with a MODE field value of SET NORMAL MODE to each port. The Master waits for the RESPONSE SMSs for CONFIGURE PORT SMS. The Master sends a MASTER ALERT SMS to each other Configutor node with an ALERT CODE value of BOTH PORTS OF A LINK ARE IN NORMAL MODE. Each Configutor node returns a RESPONSE SMS.

10.5.5 MASTER GENERATED ALERT type specific handling

The MASTER GENERATED ASYNC ALERT TYPE field value shall only be generated by the Master, and therefore the Master never receives an ASYNC ALERT SMS with a MASTER GENERATED ALERT TYPE field. If the Master receives an ASYNC ALERT SMS with a MASTER GENERATED ALERT TYPE field, then the Master performs the following actions in the order shown.

a) Build an ASYNC REPLY SMS using the TAG and ALERT CODE values from the ASYNC ALERT SMS.

b) Send the ASYNC REPLY SMS using the ADDRESS field in the ASYNC ALERT Address table entry associated with the TAG field value of the ASYNC ALERT SMS and the RETURN CODE field value of INVALID FIELD.

c) Logs the error.

A MASTER ALERT SMS is not sent to the other Configutor nodes.

10.6 MASTER ALERT SMS handling process

10.6.1 General MASTER ALERT SMS handling process

The receipt of an MASTER ALERT SMS with the following TYPE field values shall invoke general MASTER ALERT SMS Handling process.

    a)   UNKNOWN SMS causes vendor specific error handling.

    b)   INVALID SMS causes vendor specific error handling.

    c)   PROTOCOL ERROR causes vendor specific error handling.

    d)   ENVIRONMENTAL ERROR causes vendor specific error handling.

    e)   WARNING causes vendor specific error handling.

    f)   ERROR THRESHOLDS causes vendor specific error handling.

The receipt of an MASTER ALERT SMS by the Configutor node invokes the MASTER ALERT SMS Handling process defined as follows in the order shown.

    a)   Build a RESPONSE SMS using the TAG value from the MASTER ALERT SMS.

    b)   Send the RESPONSE SMS using the RETURN PATH value in the MASTER ALERT SMS as a frame ADDRESS field value.

    c)   Log the error.

10.6.2 PORT NOW OPERATIONAL type handling process

The PORT NOW OPERATIONAL Type handling process is performed in the order shown:

    a)   Invoke the Configuration process starting at the node and port that is now operational.

    b)   Perform the General MASTER ALERT SMS Handling process (see 10.6.1).

10.6.3 ADDRESS ERROR type handling process

The ADDRESS ERROR Type handling process is performed in the order shown:

    a)   Terminate any affected I/O processes and the associated out-bound data transfers.

    b)   Each Configutor node invalidates the entries in the Configuration table for the affected link(s) and node(s).

    c)   Perform the General MASTER ALERT SMS Handling process (see 10.6.1).

    d)   If there is an alternative path, reissue the affected commands.

10.6.4 LINK ERP EXITS type handling process

The LINK ERP EXITS Type handling process is performed in the order shown:

    a)   Invalidate the entries in the Configuration table for the affected link(s) and node(s).

    b)   Stop the associated out-bound data transfers of those commands that were in progress over the failing link.

    c)   Issue a QUIESCE SMS to each node that was executing an affected command, providing the node remains configured (i.e. by an alternate path).

    d)   When a RESPONSE SMS has been received for each QUIESCE SMS, a RESPONSE SMS is sent to the Master for the MASTER ALERT SMS when all of its affected nodes have been quiesced.

    e)   Issues a QUERY NODE SMS to each node that was quiesced over each path that is intended to be used. This rebuilds the Configutor table.

    f)   If the link is still configured or there is an alternative path then reissue the affected commands.

10.6.5 MASTER GENERATED ASYNC ALERT type handling process

The actions taken with the MASTER GENERATED ALERT TYPE field value shall depend on the ALERT CODE value as defined below.

10.6.5.1 ALL OPERATIONAL PORTS ON ALL NODES ARE IN NORMAL MODE

Any Configutor node receiving this MASTER ALERT SMS informs its upper level protocol(s) that all nodes are now capable of receiving Application frames, then invokes the General MASTER ALERT SMS Handling process (see 10.6.1).

10.6.5.2 ALL OPERATIONAL PORTS ON THE SPECIFIED NODE ARE IN NORMAL MODE

This MASTER ALERT SMS is for use in switch node applications. The receiving Configutor node does not inform its upper level protocol manager that the node identified is able to receive Application frames until all the port on the path to that node are in Normal mode, then invokes the General MASTER ALERT SMS Handling process (see 10.6.1).

10.6.5.3 RECONFIGURATION REQUIRED

This MASTER ALERT SMS follows the Total Reset of a Configutor node. Any Configutor node receiving this MASTER ALERT SMS clears its Configutor table and reconfigure the Web as described in 9.4, , then invokes the General MASTER ALERT SMS Handling process (see 10.6.1).

10.6.5.4 BOTH PORTS OF LINK ARE IN NORMAL MODE

Any Configutor node receiving this MASTER ALERT SMS does not inform its upper level protocol that the identified node is able to receive Application frames until all the ports on the path to that node are in Normal mode, then invokes the General MASTER ALERT SMS Handling process (see 10.6.1).

11 SSA-TL1 message structures

The typical SSA-TL1 message structure (SMS) is defined in Table 21. If the SMS CODE field value (byte 0) ≥ 80h, then the SMS is an upper level protocol SMS with the format defined in the appropriate upper level protocol. The SMS is transported in the DATA field of a frame (see 7.5).

Table 21 - Generic SSA-TL1 message structure

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE							
1	(varies with SMS)							
2	TAG							
3	TAG							
4	(varies with SMS)							
...	(varies with SMS)							
31	(varies with SMS)							

An SMS has a maximum length of 32 bytes. Only the bytes defined in the appropriate SMS structure are required, however if the SMS is padded, it shall be padded up to a total length of 32 bytes. If the source node pads the SMS, the pad bytes shall be zero. The destination node shall ignore any pad bytes.

A path address in an SMS (e.g., RETURN PATH, ALERT PATH) shall be left-aligned in a four-byte field. The EXTEND flag function is defined in 7.4.1. The bytes of the path address following the byte with an EXTEND flag of zero are ignored.

The SMS codes that are valid for SSA-TL1 and associated frame type and node type support is defined in Table 22.

Table 22 - SSA-TL1 messages supported

SMS	SMS CODE	SMS FRAME TYPE	Node type support	
			Sent from	Received by
QUERY NODE	00h	Privileged	Configutor	all
QUERY NODE REPLY	01h	Privileged	all	Configutor
CONFIGURE PORT	02h	Privileged	Master	all
RESPONSE	03h	Privileged	all	Configutor
ASYNC ALERT	04h	Privileged	all	Master
MASTER ALERT	05h	Privileged	Master	Configutor
QUIESCE	06h	Privileged	Configutor	all
ASYNC REPLY	07h	Privileged	Master	all
QUERY PROTOCOL	08h	Privileged	Configutor	all
QUERY PROTOCOL REPLY	09h	Privileged	all	Configutor
QUERY PORT	0Ah	Privileged	Configutor	all
QUERY PORT REPLY	0Bh	Privileged	all	Configutor
QUERY SWITCH	0Ch	Privileged	Configutor	all
QUERY SWITCH REPLY	0Dh	Privileged	all	all
reserved for Privileged frames	0Eh-1Fh	Privileged	reserved	reserved
DATA READY	20h	Application	all	all
DATA REPLY	21h	Application	all	all
DATA REQUEST	22h	Application	all	all
reserved for Application frames	23h-7Fh	Application	reserved	reserved
Defined by ULP	see 11.2.6	defined in ULP		

See 7.1.1 for information on encoding the frame type.

11.1 SMS validation

Each SMS undergoes three levels of validation.

11.1.1 SMS code validation

The code value of an SMS shall be validated in the order shown:

a) If the SMS CODE value is greater than or equal to 80h and the specified ULP is supported, then the SMS is sent to the specified ULP. The FRAME TYPE is not checked.

b) If the SMS CODE value is greater than or equal to 80h and the specified ULP is not supported, then an Asynchronous Alert process shall be invoked with an ALERT CODE value of UNSUPPORTED ULP.

c) If the SMS CODE value is not valid, then an Asynchronous Alert process shall be invoked with an ALERT CODE value of UNKNOWN SMS.

d) If the SMS CODE value is a non-reserved value shown in Table 22, but the FRAME TYPE or NODE TYPE are incorrect, then an Asynchronous Alert process shall be invoked with an ALERT CODE value of UNKNOWN SMS.

e) If the SMS CODE value corresponds to the CONFIGURE PORT, MASTER ALERT, QUERY PORT, QUERY PROTOCOL, or QUIESCE SMS, and one of these SMSs is already outstanding in the node from this Configutor node, then generate a RESPONSE SMS with a RETURN CODE field value of OVERLAPPED SMSS ATTEMPTED.

f) If the SMS is a QUERY NODE, and the node has no room to store the SMS, the node shall discard the QUERY NODE SMS. The node shall not generate any type of response. A Configutor node shall employ a time-out to retry the QUERY NODE if no QUERY NODE REPLY is received.

g) If Table 22 matches the SMS CODE value (excluding reserved values), the FRAME TYPE of the SMS, and the node type, then continue processing the SMS as defined in the clause 11.

11.1.2 SMS length validation

The length of an SMS shall be validated. If the number of bytes received is less than specified for the SMS the port shall invoke an Asynchronous Alert process with an ALERT CODE value of SMS TOO SHORT.

11.1.3 SMS field validation

The fields of an SMS shall be validated as described below. If the Asynchronous Alert process is invoked then the message is terminated, without any other response. The RETURN PATH field of Privileged SSA-TL1 SMSs need not be registered in the Configutor table. The SMS fields shall be validated in the order as follows:

a) If any reserved field is non-zero or a valid field contains a reserved value, then generate an appropriate response as defined in clause 10.2.7. This checking is not performed against the QUERY NODE and QUERY NODE REPLY SMSs.

b) If the TAG field is invalid, or valid with an improper response, then the Configutor node logs the error and ignores the SMS. If the SMS was a data transfer SMS, the Configutor node aborts the I/O process using the ULP if the ULP is known.

11.2 Privileged SMS

11.2.1 ASYNC ALERT SMS

The ASYNC ALERT SMS defined in Table 23 is sent by a node to inform the Master of an Asynchronous Alert. The Master forwards the ASYNC ALERT SMS to each Configutor node as a MASTER ALERT SMS (see 11.2.4). The Master returns an ASYNC REPLY SMS when recovery has been completed (see 11.2.2).

Table 23 - ASYNC ALERT SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (04h)							
1	PORT							
2	TAG							
3	TAG							
4	ALERT CODE							
5	ALERT CODE							
6	ALERT CODE							
7	reserved							
8	CONTROL							
9	CHANNEL							
10	CHANNEL							
11	FRAME DATA							
	FRAME DATA							
n	FRAME DATA							

The PORT field contains an unsigned integer to identify the affected port.

The TAG field contains the tag specified by the most-recent CONFIGURE PORT SMS for the affected port. It allows the Master to determine the affected port on the node that sent the Asynchronous Alert.

The ALERT CODE field consists of three parts as shown in Table 24.

Table 24 - Format of the ALERT CODE field

Byte within SMS	Byte Name	Description
4[1]	TYPE	The general type of Asynchronous Alert generated
5[1]	SUBTYPE	Additional information that is type dependent.
6[1]	TYPE INFORMATION	Additional information type and subtype dependent.
[1]Note: The TYPE, SUBTYPE, and TYPE INFORMATION fields in the MASTER ALERT SMS are in bytes 16, 17, and 18 respectively.		

The CONTROL field contains the control byte of the frame causing the ASYNC ALERT SMS.

The CHANNEL field contains the channel of the frame causing the ASYNC ALERT SMS.

The FRAME DATA field contains up to the first 21 bytes of the data field of the frame causing the Asynchronous Alert.

The CONTROL, CHANNEL, and FRAME DATA fields are only valid for meaning codes of UNKNOWN SMS, INVALID SMS, or PROTOCOL ERROR.

The ADDRESS field in an ASYNC ALERT SMS is copied from the RETURN PATH field in the most recent CONFIGURE PORT SMS for the affected port. Any padding bytes shall be discarded.

The acknowledgment for the ASYNC ALERT SMS is the receipt of an ASYNC REPLY SMS with the same tag value. The RETURN CODE field indicates if the ASYNC ALERT SMS was successfully parsed (see 11.2.2).

The Asynchronous Alert process shall be invoked for a Link ERP process exit if the ASYNC ALERT SMS is sent via a different port than the one that had the Link ERP process exit. If the ASYNC ALERT SMS is to be sent via the same port then the alert is discarded. The remote port is still be able to report the alert since it is nearer the Master. All other types of Asynchronous Alert are always reported.

A particular alert remains pending in the affected port until the node receives an ASYNC REPLY SMS for the corresponding ASYNC ALERT SMS. The ASYNC REPLY SMS indicates that the alert has been processed by the Master. If the node receives no ASYNC REPLY within 500 ms of sending an ASYNC ALERT SMS then it retransmits the ASYNC ALERT SMS. This cycle repeats indefinitely until an ASYNC REPLY SMS, a Total Reset, or an Absolute Reset is received. The Master shall be prepared to receive more than one ASYNC ALERT SMS for the same event and send an ASYNC REPLY SMS for each SMS. Also the affected node shall be prepared to discard an ASYNC REPLY SMS if the corresponding port has no alert pending.

The CONFIGURE PORT SMS allocates a single tag to be used by ASYNC ALERT SMS. Therefore, each port shall have only one ASYNC ALERT SMS that is awaiting an ASYNC REPLY SMS at any given time, requiring queuing of Asynchronous Alerts be supported in the port.

The generation of ASYNC ALERT SMSs is controlled by the De-queue Asynchronous Alert process (see 10.4).

The meaning code defined in Table 25 is the description assigned the combination of the TYPE, SUBTYPE, and TYPE INFO fields.

Table 25 - ALERT CODE values

TYPE	SUBTYPE	TYPE INFORMATION	ALERT CODE VALUE	Port mode
Type 00h, PORT NOW OPERATIONAL				
00h	01h	00h	PORT NOW OPERATIONAL	Privileged
Type 01h, ADDRESS ERROR				
01h	01h	00h	ADDRESSED PORT NOT OPERATIONAL	Privileged
Type 02h, UNKNOWN SMS				
02h	01h	00h	UNKNOWN SMS CODE	Unchanged
02h	02h	00h	reserved	
02h	03h	00h	SMS TOO SHORT	Unchanged
02h	04h	00h	UNSUPPORTED ULP	Unchanged
Type 03h, INVALID SMS				
03h	01h	00h	UNKNOWN RETURN PATH OR UNKNOWN RETURN PATH ID	Unchanged
03h	02h	00h	INVALID FIELD	Unchanged
Type 04h, PROTOCOL ERROR				
04h	01h	00h	SMS UNEXPECTED	Unchanged
04h	02h	00h	reserved	
04h	03h	00h	TOO MANY SMS OUTSTANDING	Unchanged
04h	04h	00h	TOO MUCH DATA RECEIVED	Unchanged
Type 05h, ENVIRONMENTAL ERROR.				
05h	01h	00h	POWER FAULT	Unchanged
05h	02h	00h	CRITICAL FAN FAILURE	Unchanged
Type 06h, WARNING				
06h	01h	00h	REDUNDANT FAN FAILURE	Unchanged
06h	02h	vendor specific	FAILURE PREDICTION THRESHOLD EXCEEDED	Unchanged
06h	03h	00h	PORT NOT OPERATIONAL DUE TO HARDWARE FAULT	Wrap
06h	04h	Master priority	MASTER PRIORITY CHANGED	Unchanged
Type 10h to 19h, Link ERP process exits. Local = Link Status Byte sent by the node originating the ASYNC ALERT. Remote = Link Status Byte received by the node originating the ASYNC ALERT.				
10h	00h	00h	PERMANENT LINE FAULT	Privileged
11h	00h	00h	NO CHARACTERS RECEIVED	Privileged
12h	00h	00h	REMOTE PORT DISABLED	Privileged
13h	Local	00h	LINK RESET FAILED	Privileged
14h	Local	Remote	RETRY LIMIT EXCEEDED	Privileged
15h	Local	Remote	HARDWARE ERROR	Privileged
16h	Local	Remote	FRAME REJECT	Privileged
17h	Local	Remote	INVALID RETRY STATUS	Privileged
18h	Local	Remote	TIME-OUT WAITING FOR DISABLED STATE	Privileged
19h	Local	Remote	TIME-OUT WAITING FOR READY STATE	Privileged
Type 1Ah, ERROR THRESHOLDS.				
1Ah	00h	00h	ALARM THRESHOLD EXCEEDED	Unchanged
Type BCh to BFh, Master generated alerts (not used in ASYNC ALERT).				
BCh	00h	00h	ALL OPERATIONAL PORTS ON ALL NODES IN THE WEB ARE IN NORMAL MODE.	Normal
BDh	00h	00h	ALL OPERATIONAL PORTS ON THE SPECIFIED NODE ARE IN NORMAL MODE.	Normal
BEh	00h	00h	RECONFIGURATION REQUIRED	N/A
BFh	00h	00h	BOTH PORTS OF LINK ARE IN NORMAL MODE.	Normal

11.2.2 ASYNC REPLY SMS

The ASYNC REPLY SMS defined in Table 26 is sent by the Master to acknowledge an ASYNC ALERT SMS.

Table 26 - ASYNC REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (07h)							
1	RETURN CODE							
2	TAG							
3	TAG							
4	ALERT CODE							
5	ALERT CODE							
6	ALERT CODE							

The RETURN CODE field indicates the result of the original ASYNC ALERT SMS. The return code values are defined in Table 27.

Table 27 - RETURN CODE field values

Value	Description
00h	ASYNC ALERT SMS SUCCESSFULLY RECEIVED
FFh	INVALID FIELD

The TAG field identifies the SMS that is being acknowledged and is copied from the original ASYNC ALERT SMS. The tag value is not released and shall be reused for the next ASYNC ALERT SMS unless changed by a subsequent CONFIGURE PORT SMS.

The ALERT CODE field consists of three parts as shown in Table 25.

The node shall ignore any ASYNC REPLY SMS with TYPE, SUBTYPE and TYPE INFORMATION fields that do not match those of the outstanding ASYNC ALERT SMS.

If the TAG and ALERT CODE fields do not match the top entry in the Queued Asynchronous Alert table, the ASYNC REPLY SMS is ignored. If the TAG and ALERT CODE fields match the top entry in the Queued Asynchronous Alert table, then that entry is removed and the De-queue Asynchronous Alert process is invoked.

11.2.3 CONFIGURE PORT SMS

The CONFIGURE PORT SMS defined in Table 28 is sent by the Master to each port on every other node in the Web during the configuration process. The Master also issues a CONFIGURE PORT SMS to return a port to Normal mode after it has processed an Asynchronous Alert that placed the port in the Privileged mode. In both cases the destination node returns a RESPONSE SMS.

Table 28 - CONFIGURE PORT SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (02h)							
1	PORT							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							
8	reserved							
9	A QUOTA							
10	B QUOTA							
11	EUDC	REFLECT	MODE			reserved		
12	reserved							
13	ALARM THRESHOLD							
14	ALARM THRESHOLD							

A CONFIGURE PORT SMS specifies the return path and the tag to be used when a node sends the Master an ASYNC ALERT SMS. It also establishes the operating parameters of the port.

The PORT field contains an unsigned integer to identify the affected port.

The TAG field is returned in the RESPONSE SMS. The same tag is also used in an ASYNC ALERT SMS if the node subsequently reports an Asynchronous Alert.

The TAG value is assigned by the Master and shall be unique among the TAG values used by other ports in the Web as indicated by the Master Asynchronous Alert table (see 9.2.2). The receipt of the resulting RESPONSE SMS shall release the TAG value for use by SMSs other than the CONFIGURE PORT SMS, and the tag value shall no longer be considered active.

The RETURN PATH field specifies the path component that shall be placed in the address field of the resulting RESPONSE SMS and any subsequent Asynchronous Alert.

The A QUOTA field specifies the frame quota that a node may originate before it is satisfied. The value of the A QUOTA field shall be greater than zero.

The B QUOTA field specifies the maximum frame quota that a node may originate for each rotation of the SAT token. The value of the B QUOTA field shall be greater than or equal to the value of the A QUOTA field. If the B QUOTA value is less than the A QUOTA value, then the CONFIGURE PORT SMS shall be ignored and a RESPONSE SMS generated to the Configutor node issuing the CONFIGURE PORT SMS with the appropriate TAG value and a RETURN CODE of INVALID FIELD.

The EUDC bit (Enable User Defined Characters) specifies how the port handles User Defined characters. If the EUDC bit is cleared, the port shall not transmit any User Defined characters and shall discard received User Defined characters. If the EUDC bit is set, the port may transmit User Defined characters, and shall forward User Defined characters.

The REFLECT bit is one of the controls affecting the REFLECTION FLAG node function (see 9.2.6). If the node is a dual port node and the REFLECT bit is set, the REFLECTION FLAG is set for the node. If the node is a dual port node with the OPERATIONAL FLAG set for both ports and the REFLECT bit is cleared, the REFLECTION FLAG is cleared for the node. The REFLECT bit shall be ignored for single port nodes or switch nodes.

The MODE field selects the port mode as defined in Table 29.

Table 29 - MODE field values

Value	Description
00b	NO CHANGE TO THE CURRENT MODE.
01b	SET WRAP MODE.
10b	SET NORMAL MODE.
11b	SET PRIVILEGED MODE.

If a node receives a CONFIGURE PORT SMS with MODE field value of NO CHANGE TO THE CURRENT MODE and the specified port has an Asynchronous Alert pending then it shall send all pending ASYNC ALERT to the Master before sending the RESPONSE SMS for the CONFIGURE PORT. A MODE field value of SET WRAP MODE sets the Wrap mode and is only valid if the node received the CONFIGURE PORT SMS via a different port than that being configured. Otherwise the node returns a RESPONSE SMS with RETURN CODE VALUE of INVALID FIELD and the port mode is unchanged. When a port enters the Wrap mode, the port also becomes not Operational (see 9.1.4.3). Once a port has been placed in Wrap mode, the Master causes the node to exit the Wrap mode on the port by sending a CONFIGURE PORT SMS via a different port with a mode field value of SET PRIVILEGED MODE. The port shall exit the Wrap mode and invoke the Beginning Communication process (see 9.4). If a port currently configured in Wrap mode receives a CONFIGURE PORT SMS via a different port with a mode field value of SET NORMAL MODE, the node returns with a RESPONSE SMS with RETURN CODE VALUE of INVALID FIELD and the port remains in the Wrap mode.

The ALARM THRESHOLD field specifies the number of times the Link ERP process is invoked before the Asynchronous Alert process is invoked with an ALERT CODE value of ALARM THRESHOLD EXCEEDED. An ALARM THRESHOLD field set to zero disables the alarm threshold function. When the LINK ERP COUNTER = ALARM THRESHOLD value the node shall issue an Asynchronous Alert. Subsequent increments to the LINK ERP COUNTER above the ALARM THRESHOLD value shall not invoke the Asynchronous Alert process. If a valid QUERY PORT SMS is received with the CLE flag set, the specified port shall set the LINK ERP COUNTER to zero and thereby restart the alarm threshold function (see 11.2.8).

A port only saves the parameters of the most-recent CONFIGURE PORT that it has received.

The acknowledgment for the CONFIGURE PORT SMS is the receipt of a RESPONSE SMS with the same TAG value as was used for the CONFIGURE PORT SMS. The RETURN CODE field of the RESPONSE SMS shall indicate whether or not the CONFIGURE PORT was processed successfully (see 11.2.14).

11.2.4 MASTER ALERT SMS

The MASTER ALERT SMS defined in Table 30 is sent from the Master to each other Configutor node for the following three reasons.

   a) To forward an ASYNC ALERT SMS or an equivalent alert detected within the Master node itself. If the alert is for a Link ERP process exit then the Configutor node quiesces any I/O processes that were using the affected port before returning a RESPONSE SMS to the Master. The Configutor node invalidates the entries in the Configuration table for the affected nodes (see 11.2.1).

   b) To indicate that both ports of a link have been put into Normal mode (e.g., at power-on or following error recovery). The Configutor node returns a RESPONSE SMS to the Master immediately.

   c) To inform other Configutor nodes that they need to invoke the Configuration process to reconfigure the Web.

Table 30 - MASTER ALERT SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE = 05h							
1	PORT							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							
8	UNIQUE ID							
9	UNIQUE ID							
10	UNIQUE ID							
11	UNIQUE ID							
12	UNIQUE ID							
13	UNIQUE ID							
14	UNIQUE ID							
15	UNIQUE ID							
16	ALERT CODE							
17	ALERT CODE							
18	ALERT CODE							
19	reserved							
20	CONTROL							
21	CHANNEL							
22	CHANNEL							
23	FRAME DATA							
24	FRAME DATA							

The PORT field identifies the port that invoked the Asynchronous Alert.

The TAG field is assigned by the Master and returned in the response from the destination Configutor node. It shall be unique among all the tags currently active from the Master.

The RETURN PATH field specifies the path component that shall be placed in the ADDRESS field of the RESPONSE SMS.

The UNIQUE ID field identifies the node that invoked the Asynchronous Alert.

The ALERT CODE field consists of three parts as shown in Table 24, and is copied from the same field of the associated ASYNC ALERT SMS.

The CONTROL field is copied from the same field in the associated ASYNC ALERT SMS.

The CHANNEL field is copied from the same field in the associated ASYNC ALERT SMS.

The FRAME DATA field is copied from the first two bytes of the same field in the associated ASYNC ALERT SMS.

The acknowledgment for the MASTER ALERT SMS is the receipt of a RESPONSE SMS with the same TAG value as was used for the MASTER ALERT SMS. The RETURN CODE field of the RESPONSE SMS shall indicate whether or not the MASTER ALERT SMS was processed successfully (see 11.2.14).

11.2.5 QUERY NODE SMS

This SMS is sent from a Configutor node to every other Operational node during the Configuration process. QUERY NODE is also used as a remote wrap test to verify the integrity of the Path.

The QUERY NODE SMS is defined in Table 31.

Table 31 - QUERY NODE SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (00h)							
1	SSA-TL1 VERSION (02h)							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							
8	UNIQUE ID							
9	UNIQUE ID							
10	UNIQUE ID							
11	UNIQUE ID							
12	UNIQUE ID							
13	UNIQUE ID							
14	UNIQUE ID							
15	UNIQUE ID							
16	DR	MA	reserved					

The SSA-TL1 VERSION field defined in Table 32 identifies the version of SSA-TL1 being used by the sender.

Table 32 - SSA-TL1 VERSION field values

Version	Description
00h	SSA-TL1 implementations before standardization
01h	SSA-IA / 95PH
02h	This standard
03h-FFh	reserved

The TAG field is returned in the QUERY NODE REPLY SMS. The TAG is assigned by the Configutor node and it shall be unique among the TAG values that are currently active from that Configutor node.

The RETURN PATH field specifies the path component that shall be placed in the Configutor table entry created in response to this QUERY NODE SMS, if an entry is created. This value is used for the address field of the resulting QUERY NODE REPLY SMS, and is used for the address field of any future application SMS that utilizes this Configutor table entry.

The UNIQUE ID field contains the Unique ID of the Configutor node that issued QUERY NODE SMS.

The DR bit (Disable Registration) controls the updating of the Configutor table in the node. If the DR bit is cleared then the node shall enter the specified RETURN PATH and UNIQUE ID into its Configutor table. If a Configutor node intends to use several alternative paths to the same node then it shall issue QUERY NODE SMS with the DR bit is cleared once over each path. If the DR bit is set the Configutor table shall not be updated.

The MA bit (Master Alive) is set when the Master invokes the Master Alive process (see 12.2), and is cleared all other times. When a node receives a QUERY NODE SMS with the MA bit set, it restarts the 10 second Master Alive timer (see 12.2).

11.2.6 QUERY NODE REPLY SMS

The QUERY NODE REPLY SMS as defined in Table 33 is returned when a QUERY NODE SMS is received. The QUERY NODE REPLY SMS is returned on the same port that received the corresponding QUERY NODE SMS.

Table 33 - QUERY NODE REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (01h)							
1	PORT							
2	TAG							
3	TAG							
4	UPPER LEVEL PROTOCOL							
5	ITF	MASTER PRIORITY			reserved			
6	TOTAL OTHER PORTS							
7	SSA-TL1 VERSION (02h)							
8	UNIQUE ID							
9	UNIQUE ID							
10	UNIQUE ID							
11	UNIQUE ID							
12	UNIQUE ID							
13	UNIQUE ID							
14	UNIQUE ID							
15	UNIQUE ID							
16	RETURN PATH ID							
17	RETURN PATH ID							
18	RETURN PATH ID							
19	RETURN PATH ID							
20	P1O	P2O	reserved					

The PORT field indicates the number of the port currently being used.

The TAG field is copied from the QUERY NODE SMS. It identifies the QUERY NODE SMS that this reply is being generated for.

The UPPER LEVEL PROTOCOL field identifies the upper-level protocol that the node shall respond to. The UPPER LEVEL PROTOCOL field shall contain a value from Table 34.

Table 34 - Upper-level protocol code values

ULP code	Protocol	Notes
00h	USE THE QUERY PROTOCOL SMS TO LIST UPPER LEVEL PROTOCOLS SUPPORTED	1
01h	SHALL RESPOND TO NO UPPER-LEVEL PROTOCOL	1
02h	Vendor specific	1,2
03h-7Fh	reserved	
80h	SSA-IA / 95SP	1,2
81h	reserved	
82h	SSA-S2P	1,2
83h	SSA-S3P	1,2
84h-FAh	reserved	
FBh-FFh	Vendor specific	2
NOTES 1 - Valid in QUERY NODE REPLY. 2 - Valid in QUERY PROTOCOL REPLY		

The Configutor table full (ITF) bit is set when there is no space left in the Configutor table to make an entry for the Configutor node that sent QUERY NODE SMS.

The MASTER PRIORITY field defines the priority of the node for becoming the Web Master. A value of zero indicates that the node is not capable of functioning as a Master. A value of one is used by a node that is not Master Capable, but wishes to participate in the Healthy Web process. Any value greater than 1 indicates the node's priority for becoming the Master. A value of two is the lowest priority and seven the highest

> NOTE 10 - A Configutor node optionally fixes its Master priority at 4 (the default priority) or provides some mechanism outside of SSA to change its Master priority dynamically.

The TOTAL OTHER PORTS field contains a value that is one less than the number of ports implemented. If this value exceeds two, the QUERY SWITCH SMS is used to retrieve a port mask for the switch.

> NOTE 11 - The TOTAL OTHER PORTS field value shall be 0 (single port), 1 (dual port) or an odd number greater than 1 (switch) as per 8.2.

The SSA-TL1 field defined in Table 32 identifies the version of SSA-TL1 being used by the sender.

The UNIQUE ID field contains the node's Unique ID.

The RETURN PATH ID contains a value created by the node and returned to the Configutor node. This field shall be used by the Configutor node in any future application SMSs that are utilizing the same path as used by the QUERY NODE SMS that caused this QUERY NODE REPLY SMS is to be generated.

The P1O bit is only valid for single port and dual port nodes, Switch nodes shall clear the P1O bit. For single port or dual port nodes, the P1O bit is set if port 1 is operational, and is cleared if port 1 is not operational.

The P2O bit is only valid for single port and dual port nodes, Switch nodes shall clear the P2O bit. For single port or dual port nodes, the P2O bit is set if port 2 is operational, and is cleared if port 2 is not operational.

The Path component of the ADDRESS field in a QUERY NODE REPLY SMS frame is copied from the RETURN PATH field in the corresponding QUERY NODE SMS. All padding bytes shall be discarded.

11.2.7 QUERY PORT SMS

The QUERY PORT SMS defined in Table 35 is sent by a Configutor node to request the error statistics of the specified port. The destination node normally returns a QUERY PORT REPLY SMS. However, an invalid field causes a RESPONSE SMS to be generated with a RETURN CODE of INVALID FIELD.

Table 35 - QUERY PORT SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (0Ah)							
1	PORT							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							
8	reserved							CLE

The PORT field contains the port of the link status report.

The TAG field is returned in the QUERY PORT REPLY SMS. The TAG is assigned by the Configutor node and it shall be unique among the TAG values that are currently active from that Configutor node.

The RETURN PATH field specifies the path component that shall be placed in the path component of the ADDRESS field of the resulting QUERY PORT REPLY SMS.

The CLE bit specifies whether the node shall clear the LINK ERP COUNTER. If this CLE bit is set, the port shall reset the counter to zero after sending the QUERY PORT REPLY SMS. If CLE bit is cleared, the port shall send the current value of the LINK ERP COUNTER without clearing it.

11.2.8 QUERY PORT REPLY SMS

The QUERY PORT REPLY SMS defined in Table 36 is sent from a node to a Configutor node in response to a QUERY PORT SMS.

Table 36 - QUERY PORT REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (0Bh)							
1	reserved							
2	TAG							
3	TAG							
4	LINK ERP ERROR COUNT							
5	LINK ERP ERROR COUNT							
6	A QUOTA							
7	B QUOTA							
8	EUDC	REFLECT	MODE		reserved			
9	reserved							
10	ALARM THRESHOLD							
11	ALARM THRESHOLD							
12	SUPPORTED SPEED							
13	SUPPORTED SPEED							
14	CURRENT SPEED							
15	CURRENT SPEED							

The TAG field is copied from the QUERY PORT SMS. It identifies the QUERY PORT SMS that this response is being generated for.

The LINK ERP ERROR COUNT field is the count of the number of times the Link ERP has been invoked for the selected port since the last power on, Total Reset Control frame, Absolute Reset Control frame or the last QUERY PORT MS with the CLE bit is set. A value of FFFFh is returned if the LINK ERP COUNTER is greater than or equal to 65 535 (the count stops incrementing at 65 535).

The A QUOTA field specifies the current frame A quota for the port set by the CONFIGURE PORT SMS for the selected port.

The B QUOTA field specifies the current frame B quota for the port set by the CONFIGURE PORT SMS for the selected port.

The EUDC bit specifies the current setting of the EUDC bit for the port set by the CONFIGURE PORT SMS for the selected port.

The REFLECT bit specifies the current setting of the REFLECT FLAG for the node.

The MODE field specifies the current port mode for the port set by the CONFIGURE PORT SMS for the selected port.

The ALARM THRESHOLD field is the current setting of the ALARM THRESHOLD set by the CONFIGURE PORT SMS for the selected port (see 11.2.3).

The SUPPORTED SPEED field is defined in Table 37. The SUP40 bit shall be cleared since SSA-TL1 does not support 40 MB/sec. The SUP20 bit shall be set.

Table 37 - SUPPORTED SPEED field

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	reserved							
1	reserved						SUP40	SUP20

The CURRENT SPEED field is defined in Table 38. The CUR40 bit shall be cleared since SSA-TL1 does not support 40 MB/sec. The CUR20 bit shall be set if the port's OPERATIONAL FLAG is set, otherwise the CUR20 bit shall be cleared.

Table 38 - CURRENT SPEED field

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	reserved							
1	reserved						CUR40	CUR20

11.2.9   QUERY PROTOCOL SMS

The QUERY PROTOCOL SMS defined in Table 39 is sent by a Configutor node to determine the upper-level protocols the destination node responds to. The destination node normally returns a QUERY PROTOCOL REPLY SMS. However, an invalid field causes a RESPONSE SMS to be generated with a RETURN CODE of INVALID FIELD.

Table 39 - QUERY PROTOCOL SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (08h)							
1	reserved							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							

The TAG field is returned in the QUERY PROTOCOL REPLY SMS. The TAG is assigned by the Configutor node and shall be unique among the TAG values that are currently active from that Configutor node.

The RETURN PATH field specifies the path component that shall be placed in the path component of the ADDRESS field of the resulting QUERY PROTOCOL REPLY SMS frame.

11.2.10   QUERY PROTOCOL REPLY SMS

The QUERY PROTOCOL REPLY SMS defined in Table 40 shall be returned a QUERY PROTOCOL SMS is received. The QUERY PROTOCOL REPLY SMS shall be returned via the same port that received the corresponding QUERY PROTOCOL. It indicates the particular upper-level protocols to that the node shall respond to.

Table 40 - QUERY PROTOCOL REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (09h)							
1	reserved							
2	TAG							
3	TAG							
4	PROTOCOL LIST							
	PROTOCOL LIST							
15	PROTOCOL LIST							

The TAG field is copied from the QUERY PROTOCOL SMS. It identifies the QUERY PROTOCOL SMS that this reply is being generated for.

The PROTOCOL LIST field contains one entry for each upper-level protocol to that the node shall respond to. Upper-level protocols are identified by ULP code values as defined in Table 34. Only values returned by QUERY PROTOCOL REPLY shall be used. The protocol list field is always 11 bytes in length and is zero filled and left justified (first byte is in byte 4 of the QUERY PROTOCOL REPLY SMS).

The entry for a standard protocol is a one-byte ULP code value.

The entry for a vendor-unique protocol consists of a one byte ULP code value followed by a 4-byte protocol identifier. The ULP code value for each protocol shall be unique. The first 3 bytes of the protocol identifier identify the original vendor of the protocol in the same way as in a Unique ID. The remaining byte is assigned by the original vendor to the particular protocol.

> NOTE 12 - a node that communicates SCSI-2 and two vendor-unique protocols returns the following 11 bytes in Protocol list:
>
> ```
> 82h                          /* SSA-S2P */
> FBh 444543h 01h              /* vendor A protocol*/
> FCh 49424Dh 03h              /* vendor B protocol*/
> ```
>
> The protocol identifier is based on the OUI (see 9.2.1). The first three bytes are an OUI, the last byte is administered by the OUI assignee. For example, ACDE.4801h is the first protocol identified by the AC-DE-48 OUI assignee.

11.2.11 QUERY SWITCH SMS

The QUERY SWITCH SMS is sent from a Configutor node to each switch operational node during the configuration process to determine the ports that are operational.

The QUERY SWITCH SMS is defined in Table 41.

Table 41 - QUERY SWITCH SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (0Ch)							
1	reserved							
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							

The TAG field is returned in the QUERY SWITCH REPLY SMS. The TAG is assigned by the Configutor node and it shall be unique among the TAG values that are currently active from that Configutor node.

The RETURN PATH field specifies the Path component that shall be placed in the Path component of the ADDRESS field of the resulting QUERY SWITCH REPLY SMS

The destination node normally returns a QUERY SWITCH REPLY SMS. If a field is invalid then the destination node returns a RESPONSE SMS with a RETURN CODE field of INVALID FIELD instead of QUERY SWITCH REPLY SMS.

11.2.12 QUERY SWITCH REPLY SMS

The QUERY SWITCH REPLY SMS as defined in Table 42 is returned when a QUERY SWITCH SMS is received. The QUERY SWITCH REPLY SMS is returned on the same port that received the corresponding QUERY SWITCH SMS.

Table 42 - QUERY SWITCH REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (0Dh)							
1	PORT							
2	TAG							
3	TAG							
4	PORT MASK							
...	PORT MASK							
19	PORT MASK						reserved	

The PORT field indicates the number of the port currently being used.

The TAG field is copied from the QUERY SWITCH SMS. It identifies the QUERY SWITCH SMS that this reply is being generated for.

The PORT MASK field is a bit vector, where a bit that is set indicates that the port is operational. Byte 4, bit 7 corresponds to port 1, byte 19, bit 2 corresponds to port 126. Byte 19, bits 1 and 0 are reserved since a switch is limited to 126 ports.

11.2.13 QUIESCE SMS

The QUIESCE SMS defined Table 43 is sent from a Configutor node to a node during error recovery to abort all I/O processes that were received from the Configutor node specified by the UNIQUE ID. The node aborts the associated I/O processes, updates the Configutor table entry and returns a RESPONSE SMS. The node does not return status for an aborted I/O process unless it has already completed.

Table 43 - QUIESCE SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (06h)							
1	RCTE	reserved						
2	TAG							
3	TAG							
4	RETURN PATH							
5	RETURN PATH							
6	RETURN PATH							
7	RETURN PATH							
8	UNIQUE ID							
9	UNIQUE ID							
10	UNIQUE ID							
11	UNIQUE ID							
12	UNIQUE ID							
13	UNIQUE ID							
14	UNIQUE ID							
15	UNIQUE ID							

When the RCTE bit (remove Configutor table entry) is set, the node shall remove the Configutor table entry associated with the UNIQUE ID field. When the RCTE bit is cleared, the node shall delete all the return paths in the Configutor table entry associated with UNIQUE ID field.

> NOTE 13 - It is recommended that the RCTE bit only be set by a Configutor node quiescing it's own I/O processes, by the Master or by the resigning Master node.

The TAG field is assigned by the Configutor node and returned in the RESPONSE SMS from the node. The TAG is assigned by the Configutor node and it shall be unique among the TAG values that are currently active from that Configutor node.

The RETURN PATH field specifies the path component that shall be placed in the ADDRESS field of the RESPONSE SMS.

The UNIQUE ID field specifies the Unique ID of the Configutor node whose I/O processes are to be aborted.

Specifying the Configutor node with a Unique ID rather than a return path allows the Master to issue a third party Quiesce on behalf of a missing Configutor node.

The node shall search its Configutor table to convert the Unique ID to a return path before selecting the I/O processes to be aborted. If the Configutor node is using multiple paths to the same node then there is more than one return path for the given Unique ID and the node shall Quiesce the I/O processes for all paths.

If a Configutor node is using multiple paths to the same node then it shall issue a QUIESCE SMS over each path that remains connected. When it receives the RESPONSE SMS for a particular path the Configutor node safely assumes that no frames remain in the Web for I/O processes issued over that path.

An ordinary QUIESCE SMS also causes the node to delete all return paths from the selected Configutor table entry. The Configutor node issues a QUERY NODE SMS with the DR bit cleared over each path that it intends to continue using. This protocol ensures that there are no stale return paths left in the Configutor table when the Web configuration changes. A third-party QUIESCE SMS causes the node to delete the entire Configutor table entry for the specified Unique ID.

The acknowledgment for the QUIESCE SMS is the receipt of a RESPONSE SMS with the same tag value as was used for the QUIESCE SMS. The RETURN CODE field of the RESPONSE SMS shall indicate whether or not the quiesce was processed successfully (see 11.2.14).

11.2.14 RESPONSE SMS

The RESPONSE SMS defined in Table 44 is used to acknowledge the CONFIGURE PORT, MASTER ALERT, and QUIESCE SMSs. In addition, in the case of invalid fields, the RESPONSE SMS is used to acknowledge the QUERY PORT, QUERY PROTOCOL and DATA REPLY SMS.

Table 44 - RESPONSE SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (03h)							
1	RETURN CODE							
2	TAG							
3	TAG							

The RETURN CODE field indicates the result of the original SMS as defined in Table 45. The meaning of a return code is not dependent on the original SMS.

Table 45 - RETURN CODE values

Type	Description
00h	REQUESTED FUNCTION WAS COMPLETED SUCCESSFULLY
01h	NO I/O PROCESS FOUND
02h	UNIQUE ID NOT FOUND
03h	INVALID RETURN PATH
04h	OVERLAPPED SMS ATTEMPTED
05h-FDh	reserved
FEh	REQUESTED FUNCTION FAILED
FFh	INVALID FIELD

The INVALID FIELD RETURN CODE value indicates that the SMS was parsed successfully but one or more fields was invalid (see 11.1, for the action taken when an SMS cannot be parsed).

Each SMS definition specifies all the values of return code that are applicable to that SMS.

The TAG field is copied from the original SMS. It identifies the SMS that is being acknowledged.

The Path component of the ADDRESS field for the RESPONSE SMS frame is copied from the RETURN PATH field in the SMS being acknowledged. All padding bytes shall be discarded.

11.3 Application SMS

The FRAME TYPE field in the CONTROL field of a frame is set to indicate an Application frame (see 7.2).

The CHANNEL address in all SMS frames is 00h.

When an SMS contains a single byte Channel as a parameter it is left-aligned in a 2-byte field, and the second byte is ignored.

11.3.1 DATA READY SMS

The DATA READY SMS defined in Table 46 is sent before the outbound transfer of data. The receiving node returns one or more DATA REPLY SMSs for each DATA READY SMS. The node optionally uses more than one DATA READY SMS to transfer all of the data for a I/O process. A DATA READY SMS is considered outstanding from the time it is sent until the last Data frame associated with it is sent.

Table 46 - DATA READY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (20h)							
1	reserved							
2	TAG							
3	TAG							
4	BYTE OFFSET							
5	BYTE OFFSET							
6	BYTE OFFSET							
7	BYTE OFFSET							
8	BYTE COUNT							
9	BYTE COUNT							
10	BYTE COUNT							
11	BYTE COUNT							
12	reserved							
13	reserved							
14	TRANSFER ID							
15	TRANSFER ID							

The TAG field contains a copy of the tag field in the ULP command SMS and allows the node to associate this SMS with the correct I/O process.

The BYTE OFFSET field indicates the byte offset of the first byte to be transferred, relative to first byte requested by the I/O process. The value of the BYTE OFFSET field shall be a multiple of eight.

The BYTE COUNT field specifies the number of bytes to be transferred. The value of the BYTE COUNT field shall be a multiple of eight unless this SMS transfers the last byte requested by the command.

The sum of the BYTE COUNT and BYTE OFFSET fields in a DATA READY SMS shall not exceed the transfer length of the I/O process. Checking for violation of this rule is optional.

> NOTE 14 - Since data maybe retransmitted, the Configutor node cannot compare the sum of the BYTE COUNT fields in all the associated DATA READY SMSs with the expected total byte count of the I/O process.

The TRANSFER ID field is a unique identifier generated by the node to identify the DATA READY SMS. The value shall be unique among all outstanding DATA READY SMSs between a given pair of nodes. The node receiving the DATA READY SMS shall copy the TRANSFER ID into all associated DATA REPLY SMSs to associate the DATA REPLY SMS to a DATA READY SMS.

The protocol for the DATA READY and DATA REPLY SMSs is in the order as follows if the DDRM bit is cleared:

a) When the node is ready to transfer data it sends a DATA READY SMS. This specifies how many bytes it currently has available and the starting offset within the data requested by the command along with a transfer ID.

b) If the BYTE COUNT, BYTE OFFSET, or TRANSFER ID do not conform to the rules above, then the receiving node aborts the I/O process as allowed by the upper level protocol. Otherwise the receiving node allocates a channel to receive the data.

c) The node receiving the DATA READY SMS responds with a DATA REPLY SMS indicating the CHANNEL and BYTE COUNT that it is willing to currently accept and copies the TRANSFER ID from the associated DATA READY SMS.

d) The node sends Data frames addressed to the specified channel.

e) If all of the data offered by the previous DATA READY SMS has not been transferred then the node returns to step 3 using the same or a different channel. The node may send a DATA REPLY SMS for an outstanding DATA READY SMS whenever there is at most one outstanding DATA REPLY SMS for that DATA READY SMS. A DATA REPLY SMS shall not span multiple DATA READY SMSs.

f) If all of the data requested by the I/O process has not been transferred then the node returns to step a.

g) The node may send a DATA READY SMS when either there are no outstanding DATA READY SMSs for the command, or, at least one DATA REPLY has been received for each DATA READY that is outstanding for the command.

h) A Data frame shall not span multiple DATA REPLY SMSs or DATA REQUEST SMSs.

11.3.2 DATA REPLY SMS

The DATA REPLY SMS defined in Table 47 is sent in response to a DATA READY SMS. The node receiving the DATA REPLY SMS sends the requested Data frames. More than one DATA REPLY SMS may be sent in response to a single DATA READY SMS. A DATA REPLY SMS is considered outstanding from the time it is sent until the last Data frame associated with it is received.

Table 47 - DATA REPLY SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (21h)							
1	reserved							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	BYTE COUNT							
9	BYTE COUNT							
10	BYTE COUNT							
11	BYTE COUNT							
12	CHANNEL							
13	CHANNEL							
14	TRANSFER ID							
15	TRANSFER ID							

The RETURN PATH ID field identifies the node that issued the DATA REPLY SMS. The RETURN PATH ID field and TAG field serve to identify the I/O process. If the RETURN PATH ID field does not match the corresponding field of the ULP command SMS, an Asynchronous Alert is generated with an ALERT CODE value of INVALID RETURN PATH

OR RETURN PATH ID. The node then waits for another DATA REPLY SMS or until the I/O process is terminated. The RETURN PATH ID field specifies the node issuing the DATA REPLY SMS, since tags may be redundant across Configutor nodes.

The BYTE COUNT field specifies the number of bytes that the node is willing to accept during the transfer associated with this DATA REPLY SMS. If it is less than the BYTE COUNT field of the corresponding DATA READY SMS, then the node shall send another DATA REPLY SMS when it is ready to accept more data. The value of the BYTE COUNT field shall be a multiple of eight unless this SMS transfers the last byte requested by the command. If the modulo eight rules regarding the BYTE COUNT field are violated, the SMS is rejected by invoking the Asynchronous Alert process with an ALERT CODE value of INVALID FIELD.

The CHANNEL field specifies the channel component of the ADDRESS field for the Data frames. The path component is obtained from the RETURN PATH field.

The DATA REPLY SMSs for a single DATA READY SMS may contain different CHANNEL values. If two consecutive DATA REPLY SMSs for the same DATA READY SMS contain the same CHANNEL value, then the node shall complete the first DATA REPLY SMS's transfer before beginning the second DATA REPLY SMS's transfer. If two consecutive DATA REPLY SMSs for the same DATA READY SMS contain different CHANNEL values, the node may overlap the data transfers.

In the case of an invalid field other than RETURN PATH ID, then the DATA REPLY SMS is acknowledged with a RESPONSE SMS with a RETURN CODE of INVALID FIELD. In the case of an invalid RETURN PATH ID field value, an SMS UNEXPECTED, or TOO MANY SMS OUTSTANDING, an ASYNC ALERT SMS is generated with the appropriate meaning codes. The associated I/O processes identified by the tag value (if any) is not terminated, instead the DATA REPLY SMS is rejected with a RESPONSE SMS or an ASYNC ALERT SMS.

11.3.3 DATA REQUEST SMS

The DATA REQUEST SMS defined in Table 48 is sent from by a node to request the transfer of the data for the specified channel. Multiple DATA REQUEST SMSs may be used to transfer the data for an I/O process. A DATA REQUEST SMS is considered outstanding from the time it is sent until the last Data frame associated with it is received.

Table 48 - DATA REQUEST SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (22h)							
1	reserved							
2	TAG							
3	TAG							
4	BYTE OFFSET							
5	BYTE OFFSET							
6	BYTE OFFSET							
7	BYTE OFFSET							
8	BYTE COUNT							
9	BYTE COUNT							
10	BYTE COUNT							
11	BYTE COUNT							
12	CHANNEL							
13	CHANNEL							

The TAG field contains a copy of the tag field in the ULP command SMS and allows the node to associate this SMS with the correct I/O process.

The BYTE OFFSET field indicates the byte offset of the first byte to be transferred, relative to first byte requested by the I/O process. The value of the BYTE OFFSET field shall be a multiple of eight.

The BYTE COUNT field specifies the number of bytes the node is currently requesting. The value of the BYTE COUNT field shall be a multiple of eight unless this SMS transfers the last byte requested by the ULP command SMS.

The sum of the BYTE COUNT and BYTE OFFSET field values in a DATA REQUEST SMS shall not exceed the transfer length of the I/O process. Checking of this rule is optional.

> NOTE 15 - Since data may be retransmitted, the Configutor node cannot compare the sum of the BYTE COUNT fields in all the associated DATA REQUEST SMSs with the expected total byte count of the I/O process.

The CHANNEL field specifies the channel component of the address field to be inserted in the Data frames. The DATA REQUEST SMSs associated with a single I/O process may contain different CHANNEL values. If two consecutive DATA REQUEST SMSs for the same I/O process contain the same CHANNEL value, then the first DATA REQUEST SMS's transfer shall be completed before beginning the second DATA REQUEST SMS's transfer. If two consecutive DATA REQUEST SMSs for the same command contain different CHANNEL values, the data transfers may overlap.

The protocol for the DATA REQUEST SMS is in the order as follows:

a) When the node is ready to receive data it allocates a channel and sends a DATA REQUEST. This specifies the node channel address, how many bytes the node is requesting and the starting offset within the data specified by the I/O process. If the BYTE COUNT or the BYTE OFFSET does not conform to the rules above then the I/O process may be aborted as specified by the upper level protocol.

b) The node receiving the DATA REQUEST SMS responds by sending Data frames addressed to the specified CHANNEL. Although write data does not need to follow the same path to the node used by the I/O process, it shall arrive at the same port.

c) If all the data specified by the I/O process has not been transferred then the node returns to step 1) after there are no outstanding DATA REQUEST SMSs for the I/O process, or at least one Data frame has been received for each DATA REQUEST SMS that is outstanding for the I/O process.

d) A Data frame shall not span multiple DATA REQUEST SMS.

12 Transport layer processes

The transport layer processes are described in the following sub clauses:

a) The Power-on reset process is defined in 9.5.3.
b) The Reset process is defined in 9.5.
c) The Beginning Communication process is defined in 9.3.
d) The Configuration process is defined in 9.4.
e) The Asynchronous Alert process is defined in 10.3.
f) The De-queue Asynchronous Alert process is defined in 10.4
g) The Master Alert Handling process is defined in 10.6.

12.1 Master negotiation process

During the configuration process, the Configutor node with the highest MASTER PRIORITY is elected to be the Master (see 11.2.6). If more than one Configutor node is set at the same highest MASTER PRIORITY, then the Configutor node with the highest Unique ID among them is elected to be the Master.

Each Configutor node compares its own Master priority with the Master priority it receives from each other Configutor node in the QUERY NODE REPLY SMS. Each Configutor node in the group with the highest MASTER PRIORITY then compares its own Unique ID with those of the other Configutor nodes in the group. The Configutor node within the group with the highest Unique ID becomes the Master. If two Webs that both contain a Configutor node are joined by a new link, then each Configutor node invokes the Configuration process for the new link. The Configutor node with the lower MASTER PRIORITY or Unique ID resigns as a Master, that is, it no longer responds to ASYNC ALERT SMSs. The Configutor node with the higher MASTER PRIORITY or Unique ID issues a CONFIGURE PORT SMS to each new port to indicate that it is assuming the role of Master. It then receives any pending or future ASYNC ALERT SMSs.

The Configutor node determines that it now has the highest MASTER PRIORITY and Unique ID of the remaining Configutor nodes as described above, then it becomes the new Master. The new Master then issues a CONFIGURE PORT SMS to each remaining registered Master Capable node to indicate that it is the Master.

When the current Master changes its MASTER PRIORITY the following actions are taken in the order as follows:

a) If the current Master determines that it needs to resign then it finishes processing any ASYNC ALERT SMSs previously received and it does not send an ASYNC REPLY SMS to any future ASYNC ALERT SMS.

b) The current Master forwards its new MASTER PRIORITY to each other Configutor node in a MASTER ALERT SMS.

c) Each other Configutor node enters the new MASTER PRIORITY into its Configuration table and returns a RESPONSE SMS.

d) If another Configutor node determines that it needs to become the new Master then it issues a CONFIGURE PORT SMS to each port in the Web. Each port returns a RESPONSE SMS. The Configutor node is then the new Master.

When a Configutor node (other than the current Master) changes its MASTER PRIORITY the following actions are taken in the order as follows:

a) The Configutor node sends the current Master an ASYNC ALERT SMS containing its new MASTER PRIORITY.

b) The current Master enters the new MASTER PRIORITY into its Configuration table and forwards it to each other Configutor node in a MASTER ALERT SMS.

c) Each other Configutor node enters the new MASTER PRIORITY into its Configuration table and returns a RESPONSE SMS.

d) If the current Master determines that it needs to resign then it finishes processing any ASYNC ALERT SMSs previously received and it does not send an ASYNC REPLY SMS to any future ASYNC ALERT SMSs.

e) If the Configutor node that changed its Master priority determines that it needs to become the new Master, then it waits to receive the ASYNC REPLY SMS before issuing a CONFIGURE PORT SMS to each port in the Web. It is then the new Master.

12.2 Master alive process

The loss of the Master is detected by all Configutor nodes performing the Master Alive process described in the order as follows:

a) The Master sends a QUERY NODE SMS with the MA bit set to the Configutor node with the next lower Unique ID every 3 seconds. If the Master is the lowest Unique ID, it sends a QUERY NODE SMS with the MA bit set to the Configutor node with the highest Unique ID.

b) When a Configutor node, that is not the Master, receives a QUERY NODE SMS with the MA bit set, it sends a QUERY NODE SMS with the MA bit set to the Configutor node with the next lowest Unique ID. If no lower Unique ID exists, it sends a QUERY NODE SMS with the MA bit set to the Configutor node with the highest Unique ID.

c) If 5 seconds elapses between receipt of QUERY NODE SMSs with the MA bit set, the Configutor node sends a QUERY NODE SMS with the MA bit set to the Master.

d) If neither the QUERY NODE REPLY SMS to the QUERY NODE SMS with the MA bit set is sent to the Master nor the QUERY NODE SMS with the MA bit set is received within 5 seconds the Configutor node invokes the Configuration process.

Annex A
(informative)
Transport layer data transfer processes

A.1 Normal read transfer process

The TARGET READ, INITIATOR READ, and READ DATA COMPLETED services are for all or part of a normal read data transfer (see Annex B). The normal read data transfer consists of four parts in the order as follows:

a) The target's protocol layer issues a TARGET READ transport service that results in one or more DATA READY SMS(s) being sent (see annex sub clause B.4.4).

b) The receipt of a DATA READY SMS triggers the initiator's protocol layer to invoke the INITIATOR READ transport service, that sends one or more DATA REPLY SMS(s) (see annex sub clause A.2).

c) The receipt of a DATA REPLY SMS triggers the target's Send Read Data process (see annex sub clause A.2).

d) Sending the last byte of data triggers the target's transport layer to invoke the Read Transfer Ended process (see annex sub clause A.3).

A.2 Send read data process

The Send Read Data process is a target only process that is triggered by the receipt of a DATA REPLY SMS. The DATA REPLY SMS is validated as described in 11.3.2 using the fields in the Outstanding Data Transfer Table entry pointed to by the TAG field of the DATA REPLY SMS. The transport layer then sends the read data as specified in the DATA REPLY SMS.

A.3 Read transfer ended process.

When the target completes a read data transfer on a given Channel it finds the entry in the Outstanding Data Transfers Table that matches the Channel. If the ARRAY OF CHANNEL field contains more than one element, then the transport layer clears that Channel element. If the ARRAY OF CHANNEL field contains only one entry, then the transport layer invokes the READ DATA COMPLETED transport service as follows and clears the entry in the Outstanding Data Transfers Table.

a) The TAG field becomes the Tag parameter.

b) The RETURN PATH ID field is used to reference the Configutor table to find the Initiator's Unique ID to return in the Unique ID parameter.

c) The status of the data transfer determines the value of the Return Code parameter (FUNCTION SUCCESSFUL or FUNCTION FAILED).

A.4 Fast read transfer process

The TARGET FAST READ and FAST READ DATA COMPLETED services are intended for read data transfers where the initiator channel is already known, and no DATA READY or DATA REPLY SMSs need be exchanged. The service is initiated by the target and needs no initiator services, as the initiator has already set up for the data transfer. The fast read data transfer consists of two parts in the order as follows:

a) The target's protocol layer issues a TARGET FAST READ transport service that results in read data being transmitted (see annex sub clause B.4.7).

b) Sending the last byte of data triggers the target's transport layer to invoke the FAST READ DATA COMPLETED transport service (see annex sub clause B.4.8).

A.5 Write transfer process

The TARGET WRITE, INITIATOR WRITE, and WRITE DATA COMPLETED services control normal write transfers. The normal write data transfer consists of three parts.

1) The target's protocol layer issues a TARGET WRITE transport service that results in one or more DATA REQUEST SMS(s) being sent (see annex sub clause B.4.9).

2) The receipt of a DATA REQUEST SMS triggers the initiator's protocol layer to invoke the INITIATOR WRITE transport service, that sends the write data (see annex sub clause B.4.1).

3) Receiving the last byte of data triggers the target's transport layer to invoke the Write Transfer Ended transport process (see annex sub clause A.6).

A.6 Write transfer ended process.

When the target completes a write data transfer on a given Channel it finds the entry in the Outstanding Data Transfers Table that matches the Channel. If the ARRAY OF CHANNEL field contains more than one element, then the transport layer clears that Channel element. If the ARRAY OF CHANNEL field contains only one entry, then the transport layer invokes the WRITE DATA COMPLETED transport service as follows and clears the entry in the Outstanding Data Transfers Table.

a) The TAG field becomes the Tag parameter.

b) The RETURN PATH ID field is used to reference the Configutor table to find the Initiator's Unique ID to return in the Unique ID parameter.

c) The status of the data transfer determines the value of the Return Code parameter (FUNCTION SUCCESSFUL or FUNCTION FAILED).

Annex B
(informative)
Transport layer services

B.1 Introduction

The transport/protocol services are those services that define the logical boundary between the transport layer and the protocol layer. The services are not meant to define a hardware/software split, nor even a dividing line between software routines.

The service is either invoked by the Transport layer and execute in the Protocol layer, or invoked in the Protocol layer and execute in the Transport layer. Since this is the Transport layer, the appropriate clauses concentrates mainly on the transport layer's function in the service.

Table B.1 shows the services involved between the transport and protocol layers, along with who invokes them and where they execute.

Table B.1 - Transport services

Invoked by	Service name	Executes in	Initiator/target
Protocol	GET UNIQUE ID	Transport	Either
Protocol	SEND SMS	Transport	Either
Transport	SMS RECEIVED	Protocol	Either
Protocol	TARGET READ	Transport	target
Transport	INITIATOR READ	Protocol	Initiator
Transport	READ DATA COMPLETED	Protocol	target
Protocol	TARGET FAST READ	Transport	target
Transport	FAST READ DATA COMPLETED	Protocol	target
Protocol	TARGET WRITE	Transport	target
Transport	INITIATOR WRITE	Protocol	Initiator
Transport	WRITE DATA COMPLETED	Protocol	target
Transport	QUIESCE	Protocol	target

The transport services are separated into five main groups:

B.1.1 SMS transfer

The SEND SMS transport service is used to transmit any SMS frame. The SMS RECEIVED and GET UNIQUE ID transport services are used when an SMS is received.

B.1.2 Normal read

The TARGET READ, INITIATOR READ, and READ DATA COMPLETED services are for all or part of a normal read data transfer.

B.1.3 Fast read

The TARGET FAST READ and FAST READ DATA COMPLETED services in intended for read data transfers where the initiator channel is already known, and no DATA READY or DATA REPLY SMSs need be exchanged. The service is initiated by the target and needs no initiator services, as the initiator has already set up for the data transfer.

B.1.4 Write

The TARGET WRITE, INITIATOR WRITE, and WRITE DATA COMPLETED services control write transfers.

B.1.5 Quiesce

The QUIESCE transport service is invoked when the target receives a QUIESCE SMS, that causes the target to invoke the Quiesce Protocol service to abort all related I/O processes. A RESPONSE SMS is then generated with a RETURN CODE field related to the Return Code parameter returned by the QUIESCE transport service.

B.2 Conventions

B.2.1 Naming conventions

The term Transport Services refers to those services between the SSA-TL1 transport layer and the protocol layer, and in following SSA-TL1 conventions the name is all letters capitalized.

The terms associated with the Service are called parameters and are first letter capitalized.

Any SMS fields referenced are indicated in all small capital letters.

B.2.2 Notations

The services are indicated with indented text as illustrated below.

    Service-name (Input parm1, [Optional input parm2] || Output parm1, [Optional Output parm2])

The service name precedes the parentheses. The parentheses enclose the parameter lists. Any parameter enclosed in square brackets ([...]) is considered optional. The double bar (||) separates the input parameters from the output parameters. The transport services have no optional parameters.

B.2.3 Unique ID vs. Return Path ID

The protocol layer uses a node's Unique ID to identify it. The concepts of Port or Path are unknown to the protocol layer. To generate an SMS with a RETURN PATH ID field, the protocol layer requests the transport layer to replace bytes 4 through 7 with the RETURN PATH ID associated with the Unique ID parameter using the Replace4-7 parameter. When an SMS is parsed and determined to contains a RETURN PATH ID field, the protocol layer determines the appropriate Unique ID by issuing the GET UNIQUE ID transport service.

The transport layer controls the Configutor table and Configuration table, and uses either Unique ID or RETURN PATH ID. The transport layer provides the conversion between RETURN PATH ID and Unique ID with the GET UNIQUE ID transport service. Since the transport layer controls the generation of RETURN PATH ID, it is indirectly responsible for the control of the use of ports and multiple paths.

B.3 Transport layer data structures

B.3.1 Configuration table

The Configuration table (see 9.7.3) resides in the Initiator, and is used to choose a port and path to a node when given it's Unique ID, and to associate a Return Path ID with a Unique ID.

B.3.2 Configutor table

The Configutor table (see 9.7.4) resides in the Initiator, and is used to choose a port and path to a node when given it's Unique ID, and to associate a Return Path ID with a Unique ID.

The Configutor table contains a deadlock flag for each node that controls the number of certain SMSs that are outstanding.

B.3.3 Outstanding data transfers table

The target transport layer uses the Outstanding Data Transfers Table during data transfers and consists of an array of fields shown in Table B.2 for each outstanding data transfer. The initiator does not use this table.

Table B.2 - Outstanding Data Transfers table entry

Field	Description
TAG	The Tag value supplied by the TARGET READ or TARGET WRITE transport services.
RETURN PATH ID	The Return Path ID chosen by the TARGET READ or TARGET WRITE transport services.
ARRAY OF CHANNEL	An array of Channel values used by the data transfer.
ARRAY OF TRANSFER ID	An array of Transfer ID values used by the data transfer.

When a TARGET READ or TARGET WRITE transport service is invoked, the transport layer constructs an entry in the Outstanding Data Transfers Table. The entry is cleared when the data transfer completes and the READ DATA COMPLETED or WRITE DATA COMPLETED transport service is invoked.

The TAG field is loaded directly from the Tag parameter of the associated transport service. The TAG field is used to construct SMSs, identify a transfer with an I/O process, and to return the Tag parameter of the READ DATA COMPLETED and WRITE DATA COMPLETED transport services.

The RETURN PATH ID field is loaded from the Return Path ID that was chosen by the transport layer to send the DATA READY or DATA REQUEST SMS. The RETURN PATH ID field is used to construct the Unique ID parameter of the READ DATA COMPLETED and WRITE DATA COMPLETED transfer services. The RETURN PATH ID field is also used to validate DATA REPLY SMSs.

The ARRAY OF CHANNEL field is an array of Channel values being used by the data transfer. An entry is made each time the node is set up to accept a Channel, and is cleared when the Channel is no longer active for that data transfer. When a data transfer completes, the table is searched by Channel to find the Tag and Unique ID to be reported in the READ DATA COMPLETE or WRITE DATA COMPLETE transport service.

The ARRAY OF TRANSFER ID field is an array of Transfer ID values being used by a read data transfer. An entry is made whenever a DATA READY SMS is sent, and is cleared when the last DATA REPLY SMS associated with the DATA READY SMS is received and validated.

B.3.4 Object definitions

The parameters for the Transport/Protocol services are as follows:

a)   The Buffer Address parameter is a memory location where data is to be stored to or retrieved from.

b)   The Byte Offset parameter is a 4 byte value indicating the offset of this data transfer based on the Buffer Address parameter.

c)   The Byte Count parameter is a 4 byte value indicating the number of bytes for a data transfer. In the case of a split read or write, the byte count field only indicates the amount of data to be transferred during this portion of the split.

d)   The Control parameter indicates the CONTROL field for or from the SMS frame. It consists of the FRAME TYPE field and optionally the RESET TYPE field (if FRAME TYPE value of CONTROL FRAME). The FSN is not part of the CONTROL parameter, and is zero filled.

e)   The Length parameter is a 1 byte count of the number of bytes in the DATA field of the frame to be sent or received.

f)   The Channel parameter is a 2 byte value indicating the initiator channel for the target to use as the channel component in the ADDRESS field of the read Data frame(s).

g)   The Return Path ID parameter is the 4 byte value used by some SMSs to identify a return path.

h)   The In Order parameter is a 1 bit flag that indicates that the data is written into the buffer space as received (In Order set to 0b), or written into the buffer space in order with the Buffer Address location containing the byte of data with an offset of zero (In order set to 1b). The In Order parameter allows the transport layer to use split data transfers even if the Protocol layer does not support splits.

i)   The Replace4-7 parameter is a flag to indicate whether the transport layer replaces bytes 4 through 7 of the SMS with the Return Path ID associated with the Unique ID parameter.

j)   The Return Code parameter is an indicator of the success or failure of the service to execute properly. The valid value of the Return Code parameter are shown in Table B.3.

k) The Start Count parameter is a 4 byte value indicating the byte count to be used for the first DATA READY, DATA REPLY, or DATA REQUEST SMS.

l) The Tag parameter is a 2 byte value corresponding to the Tag value of the I/O process.

m) The Channel parameter is a 2 byte value indicating the target channel for the initiator to use as the channel component in the ADDRESS field of the write Data frame(s).

n) The Threshold Count parameter is a 4 byte value indicating the amount of available buffer space that triggers another DATA REPLY or DATA REQUEST SMS, or the amount of data available that triggers the next DATA READY SMS.

o) The Unique ID parameter is the 8 byte value reported by a node in the UNIQUE ID field of the QUERY NODE REPLY SMS and is a globally unique identifier of a node. Each Configutor in the Web uses the same Unique ID to identify a node.

Table B.3 - Return Code parameter values

Value	Name	Meaning
00h	FUNCTION SUCCESSFUL	The requested function completed successfully
01h-FDh	reserved	
FEh	FUNCTION FAILED	The requested function failed
FFh	INVALID PARAMETER	One or more parameters or fields were invalid

B.4 The transport services

B.4.1 GET UNIQUE ID

GET UNIQUE ID (Return Path ID || Unique ID, Return Code)

The GET UNIQUE ID transport service has one input parameter (Return Path ID) and two output parameters (Unique ID and Return Code). It is initiated in the protocol layer and executes in the transport layer.

The Unique ID is obtained by the Transport layer by scanning the Return Path ID array elements of the Configutor table or Configuration table and returning the associated Unique ID.

A Return Code parameter value of FUNCTION SUCCESSFUL validates the Unique ID parameter. A Return Code parameter value of FUNCTION FAILED, indicates the Return Path ID is not in the Configutor table.

B.4.2 SEND SMS

SEND SMS (Control, Unique ID, Buffer Address, Length, Replace4-7 || Return Code)

The SEND SMS transport service input parameters are Control, Unique ID, Buffer Address, Length, and Replace4-7. The return parameter is Return Code. It is initiated by the protocol layer and runs in the transport layer.

The SEND SMS transport service requests the transport layer to create and send a frame specified by the input parameters as described in the following.

a) The CONTROL field is the control parameter with the addition of the FSN that is added by the transport layer hardware.

b) The ADDRESS field of the frame is constructed by the Transport layer by using the path within the Configuration table (initiators) or Configutor table (targets) pointed to by the Unique ID parameter, combined with a Channel of 00h. The Unique ID parameter is the destination's Unique ID. The choice of the path and port is controlled by the transport layer and is vendor specific.

c) The DATA field is the number of bytes specified by the Length parameter beginning at the address pointed to by the Buffer Address pointer.

d) If the Replace4-7 parameter is set, then the transport layer replaces bytes 4 through 7 of the SMS with the RETURN PATH ID field from the Configuration table. The Replace4-7 parameter is only valid for initiator transport layers.

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the proper sending of the frame (ACK received), and not its successful receipt and parsing. A recovered error, such as a successful retry during Link ERP is considered successful and results in a FUNCTION SUCCESSFUL Return Code value. A Return Code parameter value of FUNCTION FAILED is used when the SMS was unable to be sent, such as not having that Unique ID in the Initiator or Configuration tables, or the appropriate port is non operational.

B.4.3 SMS RECEIVED

SMS RECEIVED (Control, Buffer Address, Length)

The SMS RECEIVED transport service input parameter are Control, Buffer Address, and Length. There are no output parameters. It is initiated by the Transport layer when an SMS is received over the link and passes the Transport layer SMS Validation process (see 11.1). The SMS RECEIVED process executes in the Protocol layer.

The SMS RECEIVED transport service supplies the Protocol layer with the information it needs to process the SMS as follows:

a)   The CONTROL field is the control parameter.

b)   The DATA field is the number of bytes specified by the Length parameter beginning at the address pointed to by the Buffer Address pointer.

c)   The length of the DATA field is the Length parameter.

Any errors in receiving an SMS are handled by the transport layer and are not surfaced to the protocol layer (i.e. they do not cause the SMS RECEIVED service to be invoked).

B.4.4 TARGET READ

TARGET READ (Unique ID, Tag, Buffer Address, Byte Offset, Byte Count, Start Count, Threshold Count || Return Code)

The TARGET READ transport service input parameters are. Unique ID, Tag, Buffer Address, Byte Offset, Byte Count, Start Count, Threshold Count. The return parameters are Return Code. It is invoked by the Protocol layer and executes in the Transport layer.

The TARGET READ transport service instructs the transport layer to initiate a data transfer by generating a DATA READY SMS as follows:

a)   The CONTROL field of the frame is 00h.

b)   The ADDRESS field of the frame is constructed by the Transport layer by using the path within the Configutor table pointed to by the Unique ID parameter, combined with a Channel of 00h. The choice of the path and port is controlled by the transport layer and is vendor specific.

c)   The SMS CODE field is 20h.

d)   The TAG, BYTE OFFSET, and START COUNT fields are the Tag, Byte Offset, and Byte Count parameters respectively.

e)   The TRANSFER ID field is controlled by the Transport layer and varies with the number of outstanding data transfers.

If the Start Count parameter is less than the Byte Count parameter, then additional DATA READY SMSs are generated according to the rules specified in 11.3.1 in the order as follows:

a)   Wait until the buffer has the amount of data specified in the Threshold Count parameter.

b)   The Byte Count parameter is set to the greater of zero or the Byte Count parameter decremented by the Start Count parameter.

c)   The Start Count parameter is set to the lesser of the Byte Count or Threshold Count parameters.

d)   A DATA READY SMS is generated as described previously in this clause.

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the DATA READY SMSs were sent successfully. A Return Code parameter value of FUNCTION FAILED is indicates the SMS transfer was incomplete. Possible reasons for being incomplete are Link ERP exit errors, or parameter errors.

An entry is made in the Outstanding Data Transfers Table consisting of the Tag parameter, The Return Path ID chosen by the transport layer (based on the Unique ID), all the Channel values chosen by the transport layer, and all the Transfer ID values chosen by the transport layer.

B.4.5 INITIATOR READ

INITIATOR READ (Tag, Byte Offset, Byte Count || Return Path ID, Buffer Address, Start Count, Threshold Count, In Order, Return Code)

The INITIATOR READ service input parameters are Tag, Byte Offset, and Byte Count. The output parameters are Buffer Address, Start Count, Threshold Count, In Order, and Return Code. The service is invoked by the transport layer when a DATA READY SMS is received, and runs in the protocol layer.

The INITIATOR READ service requests the initiator protocol layer to allocate buffer space for the read data, and prepare to receive the data. The initiator allocates buffer space based on the Tag, Byte Offset, and Byte Count. The initiator protocol layer then returns the buffer segment length (Start Length), the amount of buffer that shall be empty before the service issues another DATA REPLY SMS (Threshold Length), a flag to tell the service to store the data in order in the buffer (In Order), and a Return Code.

A Return Code value of FUNCTION SUCCESSFUL instructs the service to set up the hardware to transfer the data using the Return Path ID to determine the path, and the Channel from the DATA REQUEST SMS. A Return Code value of FUNCTION FAILED instructs the service to ignore the DATA REQUEST SMS and transmit no data. After returning a FUNCTION FAILED Return Code, it is the responsibility of the protocol layer to terminate the I/O process.

The INITIATOR READ transport service instructs the transport layer to set up hardware for a data transfer and generate a DATA REPLY SMS as follows:
  a)   The CONTROL field of the frame is 00h.
  b)   The ADDRESS field of the frame is constructed by the Transport layer by using the path and port within the Configutor table pointed to by the Return Path ID parameter, combined with a Channel of 00h. The choice of the path and port is controlled by the transport layer and is vendor specific.
  c)   The SMS CODE field is 21h.
  d)   The RETURN PATH ID field is the Return Path ID parameter.
  e)   The BYTE COUNT field is the Start Count parameter.
  f)   The CHANNEL field is controlled by the transport layer based on what buffer pointer set was used to receive the read data.
  g)   The TRANSFER ID field from the TRANSFER ID field on the DATA READY SMS.

If the Start Count parameter is less than the Byte Count parameter, then additional DATA REPLY SMSs are generated according to the rules specified in 11.3.1 in the order as follows:
  a)   Wait until buffer has the amount of space indicated by the Threshold Count parameter.
  b)   The Byte Count parameter is set to the greater of zero or the Byte Count parameter decremented by the Start Count parameter.
  c)   The Start Count parameter is set to the lesser of the Byte Count or Threshold count parameters.
  d)   A DATA REPLY SMS is generated as described previously in this clause.

If the In Order parameter is set, the transport layer arranges for the data to appear in the buffer in the order it was originally written to the target. This means that the data is stored beginning at the Start Address plus the Byte Offset parameter values.

B.4.6 READ DATA COMPLETED

READ DATA COMPLETED (|| Tag, Unique ID, LUN, Return Code)

The READ DATA COMPLETED service has no input parameters and has four output parameters (Tag, Unique ID, LUN, and Return Code).

The READ DATA COMPLETED service is initiated by the Read Transfer Ended transport process. The Tag, Unique ID, and LUN parameters identify the read data transfer that completed.

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the proper sending of the read Data frames (ACK received), and not its successful receipt and parsing. A recovered error, such as a successful retry during Link ERP is considered successful and results in a FUNCTION SUCCESSFUL Return Code value. A Return Code parameter value of FUNCTION FAILED is used when the SMS was unable to be sent, such as not having that Unique ID in the Initiator or Configuration tables, or the appropriate port is non Operational.

B.4.7 TARGET FAST READ

TARGET FAST READ (Unique ID, Byte Count, Channel || Return Code)

The TARGET FAST READ transport service input parameters are Unique ID, Byte Count, and Channel. The return parameter is Return Code.

The service in invoked by the protocol layer and executes in the transport layer. The TARGET FAST READ transport service then constructs a set of Data frames (128 bytes maximum each) to transfer the number of bytes in the Byte Count parameter as follows:

a) The CONTROL field of the frame is 00h.

b) The ADDRESS field of the frame is constructed by the Transport layer by using the path within the Configutor table pointed to by the Unique ID parameter, combined with a Channel of the Channel parameter.

B.4.8 FAST READ DATA COMPLETED

FAST READ DATA COMPLETED (|| Channel, Unique ID, Return Code)

The READ DATA COMPLETED service has no input parameters and has two output parameters (Channel and Return Code).

The READ DATA COMPLETED service is initiated by the transport layer when the data transfer is concluded for the operation identified by the Channel and Unique ID parameters.

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the proper sending of the read Data frames (ACK received), and not its successful receipt and parsing. A recovered error, such as a successful retry during Link ERP is considered successful and results in a FUNCTION SUCCESSFUL Return Code value. A Return Code parameter value of FUNCTION FAILED is used when the SMS was unable to be sent, such as not having that Unique ID in the Initiator or Configuration tables, or the appropriate port is non Operational.

B.4.9 TARGET WRITE

TARGET WRITE (Unique ID, Tag, Buffer Address, Byte Offset, Byte Count || Return Code)

The TARGET WRITE transport service input parameters are Unique ID, Tag, Buffer Address, Byte Offset, and Byte Count. The return parameters are Return Code. The service is invoked by the protocol layer and executes in the transport layer.

The TARGET WRITE transport service sets up the node to accept the number of bytes of data specified by the Byte Count parameter to be stored in the buffer starting at the address specified in the Buffer Address parameter. It assigns a Channel value to be used in the DATA READY SMS and to route the data.

The TARGET WRITE transport service constructs a DATA READY SMS as follows:

a) The CONTROL field of the frame is 00h.

b) The ADDRESS field of the frame is constructed by the transport layer by using the path component from the Configutor table pointed to by the Unique ID parameter, and the channel component of 00h. The choice of the path and port is controlled by the transport layer and is vendor specific.

c) The SMS CODE field is 22h.

d) The TAG field is the Tag parameter.

e) The BYTE OFFSET field is the Byte Offset parameter.

f) The BYTE COUNT field is the Byte Count parameter.

g) The CHANNEL field is controlled by the transport layer (depends on the number of outstanding data transfers and the number of channels supported by the target).

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the DATA REQUEST SMS was sent successfully. A Return Code parameter value of FUNCTION FAILED is indicates the SMS transfer was incomplete. Possible reasons for being incomplete are Link ERP exit errors, or parameter errors.

An entry is made in the Outstanding Data Transfers Table consisting of the Tag parameter, The Return Path ID chosen by the transport layer (based on the Unique ID), all the Channel values chosen by the transport layer, and all the Transfer ID values chosen by the transport layer.

B.4.10 INITIATOR WRITE

INITIATOR WRITE (Tag, Byte Count, Byte Offset || Unique ID, Buffer Address, Return Code)

The INITIATOR WRITE transport service input parameters are Tag, Byte Count, and Byte Offset. The return parameters are Buffer Address and Return Code. The service is invoked by the transport layer and executes in the protocol layer.

The INITIATOR WRITE transport service is invoked by the transport layer with the receipt of a DATA REQUEST SMS, and requests the initiator protocol layer to instruct the physical layer as to the buffer location of the data. If a buffer underrun condition occurs, the transport layer does not send any frames until sufficient data is available.

Upon receipt of a DATA REQUEST SMS, the transport layer loads the TAG, BYTE COUNT and BYTE OFFSET fields of the DATA REQUEST SMS into the Tag Byte Count, and Byte Offset parameters and invokes the INITIATOR WRITE transport service.

A Return Code parameter value of FUNCTION SUCCESSFUL instructs the transport service to set up the node to send the number of bytes of data specified by the Byte Count parameter from the buffer starting at the address specified in the Buffer Address parameter. The Data frames shall be constructed as follows:

a) The CONTROL field of the frame is 00h.

b) The ADDRESS field of the frame is constructed by the transport layer by using the path component from the Configutor table pointed to by the Unique ID parameter, and the channel component of the value specified in the CHANNEL field of the DATA REQUEST SMS. The choice of the path and port is controlled by the transport layer and is vendor specific.

c) The DATA field is constructed from the data being sent.

A RETURN CODE value of FUNCTION FAILED instructs the service to ignore the DATA REQUEST SMS and transmit no data. After returning a FUNCTION FAILED Return Code, it is the responsibility of the protocol layer to terminate the I/O process.

B.4.11 WRITE DATA COMPLETED

WRITE DATA COMPLETED (|| Tag, Unique ID, LUN, Return Code)

The WRITE DATA COMPLETED service has no input parameters and has four output parameters (Tag, Unique ID, LUN, and Return Code).

The WRITE DATA COMPLETED service is initiated by the Write Transfer Ended transport process. The Tag, Unique ID, and LUN parameters identify the read data transfer that completed.

A Return Code parameter value of FUNCTION SUCCESSFUL indicates the proper receipt of the write Data frames. A recovered error, such as a successful retry during Link ERP is considered successful and results in a FUNCTION SUCCESSFUL Return Code value. A Return Code parameter value of FUNCTION FAILED is used when an error occurred during data transfer such as too much data received, or the Link ERP failed.

B.4.12 QUIESCE

QUIESCE (Source ID || Return Code)

The QUIESCE transport service input parameter is Source ID. The return parameter is Return Code. The service is invoked by the transport layer and executes in the protocol layer.

The QUIESCE transport service is invoked by the transport layer with the receipt of a QUIESCE SMS, and instructs the Initiator to terminate all I/O processes issued by the return path identified by the Source ID parameter. The transport layer then generates a RESPONSE MESSAGE as follows:

a) The CONTROL field of the frame is 00h.

b) The ADDRESS field of the frame is constructed by the transport layer by using the path component from the RETURN PATH field from the QUIESCE SMS, and the channel component of 00h. The port is the same port that received the QUIESCE SMS.

c) The SMS CODE field is 03h.

d) The RETURN CODE field is the Return Code parameter.

Annex C
(informative)
Examples of half and full duplex

The following examples illustrate the link protocol. For compactness, the diagrams show a 2-byte ADDRESS field and a 4-byte DATA field. Each port has a pair of A/B transmit frame buffers and a pair of A/B receive frame buffers. It is also assumed that the source/destination process is fast enough to fill/empty one buffer while the link empties/fills the other. Thus frames may be transmitted back-to-back, without any intervening delay.

C.1 Half duplex

A half-duplex transfer with port 1 acting as a source and port 2 acting as a destination is shown in Figure C.1. Port 2 sends an RR character pair as soon as it detects the start of a frame since its receiver has buffer space for another frame. This clears the WAITING FOR RR FLAG in port 1 and allows another frame to be sent immediately after the trailing FLAG of the current frame. In addition, port 2 issues an ACK character pair immediately when it detects the trailing FLAG of each frame.
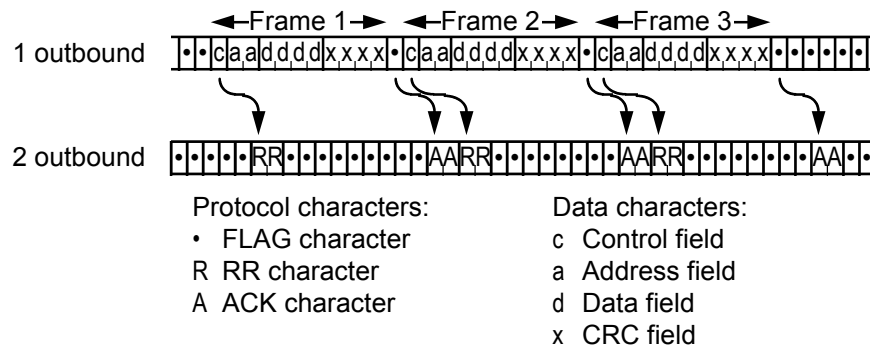

Protocol characters:
- • FLAG character
- R RR character
- A ACK character

Data characters:
- c Control field
- a Address field
- d Data field
- x CRC field

Figure C.1 - Half-duplex transfer

C.2 Full duplex

A full-duplex transfer with port 1 acting as a source of 3 frames and port 2 acting as a source of 2 frames is shown in Figure C.2. This illustrates how responses are interleaved within a frame.

Protocol characters:
- • FLAG character
- R RR character
- A ACK character

Data characters:
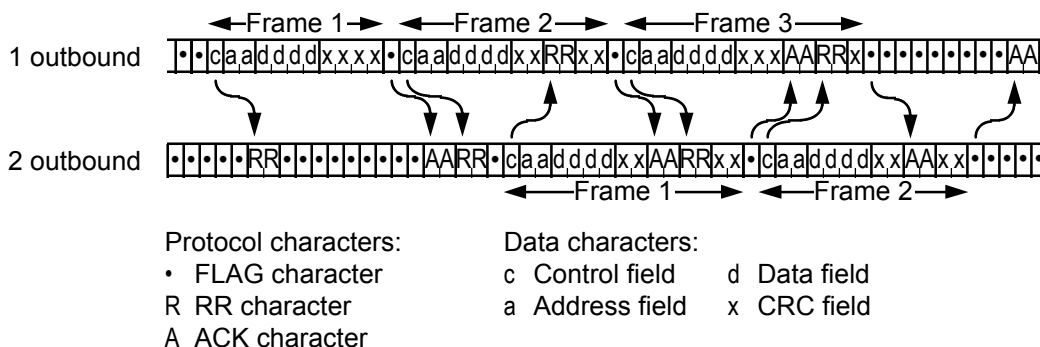- c Control field   d Data field
- a Address field   x CRC field

Figure C.2 - Full-duplex transfer