Dialogic

Dialogic® Converged Services Platform Release 8.4.1 Engineering Release 3

Developer's Guide: Common Channel Signaling

February 2008

Copyright and Legal Disclaimer

All contents of this document are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation or its subsidiaries. Reasonable effort is made to ensure the accuracy of the information contained in the document. However, due to ongoing product improvements and revisions, Dialogic Corporation and its subsidiaries do not warrant the accuracy of this information and cannot accept responsibility for errors or omissions that may be contained in this document.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH DIALOGIC® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS EXPLICITLY SET FORTH BELOW OR AS PROVIDED IN A SIGNED AGREEMENT BETWEEN YOU AND DIALOGIC, DIALOGIC ASSUMES NO LIABILITY WHATSOEVER, AND DIALOGIC DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF DIALOGIC PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF A THIRD PARTY.

Dialogic products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

It is possible that the use or implementation of any one of the concepts, applications, or ideas described in this document, in marketing collateral produced by or on web pages maintained by Dialogic Corporation or its subsidiaries may infringe one or more patents or other intellectual property rights owned by third parties. Dialogic Corporation or its subsidiaries do not provide any intellectual property licenses with the sale of Dialogic products other than a license to use such product in accordance with intellectual property owned or validly licensed by Dialogic Corporation or its subsidiaries. More detailed information about such intellectual property is available from Dialogic Corporation's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. The software referred to in this document is provided under a Software License Agreement. Refer to the Software License Agreement for complete details governing the use of the software.

Dialogic Corporation encourages all users of its products to procure all necessary intellectual property licenses required to implement any concepts or applications and does not condone or encourage any intellectual property infringement and disclaims any responsibility related thereto. These intellectual property licenses may differ from country to country and it is the responsibility of those who develop the concepts or applications to be aware of and comply with different national license requirements.

Dialogic, Dialogic Pro, Brooktrout, Cantata, SnowShore, Eicon, Eicon Networks, Eiconcard, Diva, SIPcontrol, Diva ISDN, TruFax, Realblocs, Realcomm 100, NetAccess, Instant ISDN, TRXStream, Exnet, Exnet Connect, EXS, ExchangePlus VSE, Switchkit, N20, Powering The Service-Ready

Dialogic Product Line Warranty

Unless otherwise stated in an applicable product purchase agreement between the Customer and Dialogic, Dialogic warrants that during the Warranty Period, products will operate in substantial conformance with Dialogic's standard published documentation accompanying the product. If a product does not operate in accordance therewith during the Warranty Period, the Customer must promptly notify Dialogic. Dialogic, at its option, will either repair or replace the product without charge. The Customer has the right, as their exclusive remedy, to return the product for a refund of purchase price or license fee if Dialogic is unable to repair or replace it.

Warranty Period

In the event that you have no signed agreement setting out a warranty period, the Warranty Period shall be the standard warranty period set out on www.dialogic.com on the date of your purchase of the product.

The Warranty Period begins on the date of shipment of any products or software by Dialogic.

The Warranty Period for repaired, replaced or corrected products and software shall be coterminous to the Warranty Provided for the original products or software purchased.

To report warranty claims, Customer may contact Dialogic via email at techsupport@cantata.com or call (781) 433-9600.

Warranty Provisions

A. During the Warranty Period, Dialogic warrants to Customer only that:

(i) Products manufactured by Dialogic (including those manufactured for Dialogic by an original equipment manufacturer) will be free from defects in material and workmanship and will substantially conform to specifications for such products;

(ii) software developed by Dialogic will be free from defects which materially affect performance in accordance with the specifications for such software. With respect to products or software or partial assembly of products furnished by Dialogic but not manufactured by Dialogic, Dialogic hereby assigns to Customer, to the extent permitted, the warranties given to Dialogic by its vendors of such items.

B. If, under normal and proper use, a defect or non conformity appears in warranted products or software during the applicable Warranty Period and Customer promptly notifies Dialogic in writing during the applicable warranty period of such defect or non conformance, and follows Dialogic's instructions regarding return of such defective or non conforming Product or Software, then Dialogic will, at no charge to Customer, either:

(i) repair, replace or correct the same at its manufacturing or repair facility or

(ii) if Dialogic determines that it is unable or impractical to repair, replace or correct the product or software, provide a refund or credit not to exceed the original purchase price or license fee.

C. No product or software will be accepted for repair or replacement without the written authorization of and in accordance with instructions from Dialogic. Removal and reinstallation expenses as well as transportation expenses associated with returning such product or software to Dialogic shall be borne by Customer. Dialogic shall pay the costs of transportation of the repaired or replaced product or software to the destination designated in the original Order. If Dialogic determines that any returned product or software is not defective, Customer shall pay Dialogic's costs of handling, inspecting, testing and transportation. In repairing or replacing any product, part of product, or software medium under this warranty, Dialogic may use new, remanufactured, reconditioned, refurbished or functionally equivalent products, parts or software media. Replaced products or parts shall become Dialogic's property.

D. Dialogic makes no warranty with respect to defective conditions or non conformities resulting from any of the following: Customer's modifications, misuse, neglect, accident or abuse; improper wiring, repairing, splicing, alteration, installation, storage or maintenance performed in a manner not in accordance with Dialogic's or its vendor's specifications, or operating instructions; failure of Customer to apply Dialogic's previously applicable modifications or corrections; or items not manufactured by Dialogic or purchased by Dialogic pursuant to its procurement specifications. Dialogic makes no warranty with respect to products which have had their serial numbers removed or altered; with respect to expendable items, including, without limitation, fuses, light bulbs, motor brushes and the like; or with respect to defects related to Customer's data base errors. Improper packaging of product for repair will not be covered under this warranty agreement. No warranty is made that software will run uninterrupted or error free.

E. Warranty does not include:

a) Dialogic's assistance in diagnostic efforts;

b) access to Dialogic's Technical Support web sites, databases or tools;

c) product integration testing;

d) on-site assistance; or

e) product documentation updates.

These services are available either during or after warranty at Dialogic's published prices.

F. THE FOREGOING WARRANTIES ARE EXCLUSIVE & ARE GRANTED IN LIEU OF ALL OTHER EXPRESS & IMPLIED WARRANTIES (WHETHER WRITTEN, ORAL, STATUTORY OR OTHERWISE), INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. CUSTOMER'S SOLE AND EXCLUSIVE REMEDY AND DIALOGIC'S SOLE OBLIGATION HEREUNDER, SHALL BE TO REPAIR, REPLACE, CREDIT OR REFUND AS SET FORTH ABOVE.

G. IN NO EVENT SHALL DIALOGIC, ITS DIRECTORS, OFFICERS, EMPLOYEES, AGENTS OR AFFILIATES, BE LIABLE FOR ANY COSTS OR DAMAGES ARISING DIRECTLY OR INDIRECTLY FROM YOUR USE OF ANY PRODUCT INCLUDING ANY INDIRECT,

INCIDENTAL, SPECIAL, EXEMPLARY, MULTIPLE, PUNITIVE OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, WHETHER BASED ON CONTRACT, TORT (INCLUDING NEGLEGENCE), STRICT LIABILITY OR OTHER LEGAL THEORY, EVEN IF DIALOGIC, OR ANY OF ITS DIRECTORS, OFFICERS, EMPLOYEES, AGENTS OR AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY EVENT, DIALOGIC'S CUMULATIVE LIABILITY TO YOU FOR ANY AND ALL CLAIMS RELATING TO THE USE OF ANY PRODUCT SHALL NOT EXCEED THE TOTAL AMOUNT OF THE PURCHASE PRICE OR LICENSE FEES PAID TO DIALOGIC FOR SUCH PRODUCT.

H. CUSTOMER AND DIALOGIC HEREBY WAIVE THEIR RIGHT TO TRIAL BY JURY TO THE FULLEST EXTENT PERMITTED BY LAW IN CONNECTION WITH ALL CLAIMS ARISING OUT OF OR RELATED TO THIS WARRANTY, THE PRODUCTS COVERED HEREBY OR THE PERFORMANCE OF ANY PARTY HEREUNDER.

I. THIS WARRANTY SHALL BE CONSTRUED UNDER AND GOVERNED BY THE LAWS OF THE COMMONWEALTH OF MASSACHUSETTS WITHOUT GIVING EFFECT TO ANY CHOICE OR CONFLICT OF LAW PROVISION OR RULE (WHETHER OF THE COMMONWEALTH OF MASSACHUSETTS OR ANY OTHER JURISDICTION) THAT WOULD CAUSE THE APPLICATION OF THE LAWS OF ANY JURISDICTION OTHER THAN THE COMMONWEALTH OF MASSACHUSETTS. CUSTOMER SPECIFICALLY AND IRREVOCABLY CONSENTS TO THE PERSONAL AND SUBJECT MATTER JURISDICTION AND VENUE OF THE FEDERAL AND STATE COURTS OF THE COMMONWEALTH OF MASSACHUSETTS AND SUCH COURTS SHALL HAVE EXCLUSIVE JURISDICTION WITH RESPECT TO ALL MATTERS CONCERNING THIS WARRANTY OR THE ENFORCEMENT OF ANY OF THE FOREGOING.

J. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM JURISDICTION TO JURISDICTION.

About this Publication

Purpose

This documentation provides guidelines for using the Dialogic® CSP.

Safety Labels

The following Safety labels may appear in this information product to alert customers to avoidable hazards. The following are in the order of priority:

DANGER

Danger indicates the presence of a hazard that will cause death or severe personal injury if the hazard is not avoided.

WARNING

Warning indicates the presence of a hazard that can cause death or severe personal injury if the hazard is not avoided.

CAUTION

Caution indicates the presence of a hazard that will or can cause minor personal injury or property damage if the hazard is not avoided. Caution can also indicate the possibility of data loss, loss of service, or that an application will fail.

Conventions used

This information product uses the text conventions explained below. In addition, hexadecimal numbers are preceded by a zero and small "x." For example, the decimal number 15 is represented in hexadecimal as 0x0F.

Convention	Description
. . .	A horizontal ellipsis in an API message indicates fields of variable length.
:	A vertical ellipsis in an API message indicates that a block of information is repeated or is variable.
n	The letter *n* is a generic placeholder for a number.
`Sans serif mono space`	Indicates a command name, option, input, output, non-GUI error, and system messages.
`Sans serif monospace italic`	Indicates a parameter name in an input message. Example: `move *.dot a: c:` *`-s`* The *`-s`* is the parameter.
Serif italic	Indicates the name of a book, chapter, path, file, or API message. Example: *`UserDirectory/Config.exe`*
Boldface	Indicates keyboard keys, key combinations, and command buttons Example: **Ctrl+Alt+Del**
Sans serif boldface	Identifies text that is part of a graphical user interface (GUI). Example: Go to the **Configuration** menu and select **Card**->**Span Configuration**

Contents

5 TUP, BT IUP & SSUTR2

6 SCCP/TCAP

7 SS7 PPL Information

A Appendix

1 Common Channel Signaling

Purpose

In a multi-channel communications system, Common Channel Signaling (CCS) is signaling in which one channel in each link is used for signaling to control, account for, and manage traffic on all channels of the link. The channel used for common-channel signaling does not carry user information. With CCS, signals are carried by a separate network consisting of Signaling Data Links (SDLs) and Signaling Transfer Points (STPs) to transfer digital signaling messages between exchanges. It is known as CCS because SDLs can carry messages for trunks in many different trunk groups.

Supported Protocols

The CSP supports the following CCS protocols:

- Signaling System 7
- Integrated Services Digital Network
- Q Signaling (QSIG)
- DASS2/DPNSS
- V5.2 (Local Exchange side)

SS7

A telecommunications network served by common channel signaling is composed of a number of switching and processing nodes interconnected by transmission links. To communicate using SS7, each of these nodes must implement the necessary “within node” features of SS7 making that node a signaling point within the SS7 network. In addition, there is a need to interconnect these signaling points such that SS7 signaling data may be conveyed between them. These data links are the signaling links of the SS7 signaling network.

The combination of signaling points and their interconnecting signaling links form the SS7 signaling network.

ISDN The main feature of Integrated Services Digital Network (ISDN) is the support of a wide range of service capabilities, including voice and non-voice applications, in the same network by offering end-to-end digital connectivity. A key element of service integration for an ISDN is the provision of a limited set of standard multi-purpose user-network interfaces. These interfaces represent a focal point both for the development of ISDN network components and configurations and for the development of ISDN terminal equipment and applications.

QSIG The CSP supports the QSIG/PSS1 global signaling and control standard for Private Integrated Network Exchange (PINX) applications, intended for use in private corporate ISDN networks. QSIG is a Euro-ISDN based protocol for digital Common Channel Signaling (CCS) and is used to build private networks using Virtual Private Networks (VPNs) or leased lines.

Q Signaling (QSIG), an ISDN based protocol, enables signaling between different voice communications platforms and equipment (nodes) in a multi-user environment. It is often referred to as an inter-PBX signaling system. It can also be deployed in a single-user environment.

Internationally, QSIG is also known as Private Signaling System No. 1 (PSS1).

DASS2 DASS2 is a message-based signaling system following the ISO-based model developed by British Telecom to provide multi-line Integrated Digital Access (IDA) interconnection to the BT network.

DPNSS Digital Private Network Signaling System is a standard in Britain which enables Private Branch Exchanges (PBXs) from different manufacturers to be tied together with E1 lines and pass calls transparently between each. The calls are made as easily as if the phones were extensions off the same PBX and were simply intercom calls.

V5.2 V5.2 is a concentration protocol for digital Common Channel Signaling (CCS). It is called a concentration protocol because it can accommodate more subscribers than existing physical ports. The V5.2

protocol is comprised of the LE (Local Exchange) side of the protocol. The V5.2 protocol is used to establish, maintain, and release calls between an LE and an AN.

Dialogic's V5.2 products include the hardware and software required to run and manage V5.2 on the Converged Services Platform (CSP). This implementation supports only the Local Exchange (LE) side of V5.2.

2 Introduction to SS7

Purpose

This chapter will familiarize you with Signaling System 7 (SS7) and the operation of the SS7 network. This chapter describes the CSP implementation of SS7 and provides concepts and information necessary to use the SS7 PQ and SS7 Series 3 cards. For a list of the common acronyms used in SS7, see the Appendix of this book.

The SS7 PQ and SS7 Series 3 cards provide Signaling System 7 connectivity for CSPs, and also provides a fully integrated solution for both switching and advanced signaling applications.

In this guide, where reference is made to “SS7 cards”, the reference is for both the SS7 PQ card and the SS7 Series 3 card.

Basics of SS7

Overview

Signaling System 7 SS7 is a network protocol that provides control for telecommunications networks. SS7 achieves this control by creating and transferring the following tasks to various network components:

- Call Processing
- Network management
- Maintenance

The SS7 cards allows the CSP to act as a Signaling Point (SP) in the SS7 network architecture.

The CSP generates and receives SS7 messages to manage voice circuits and provide a transport mechanism for Advanced Intelligent Network (AIN) applications.

Licenses for SS7 Software Modules

A Product License Key is required for SS7 cards. On the SS7 card, the lower level software modules are included in the price of the card, but others must be licensed at additional cost. You can buy additional product licenses for the following SS7 User Part features, which run on the SS7 card:

- ISDN User Part (ISUP) including ANSI, Japan, and ITU-T variants
- Telephone User Part (TUP) Blue Book version. China TUP requires a special PPL protocol

British Telecommunications version of CCITT SS7 Interconnect User Part (BT-IUP) ANSI and ITU-T variants of ISUP can run simultaneously on a single card. A single SS7 card can run either ISUP (and multiple variants) and/or TUP.

You can also buy additional product licenses for the following SS7 features, which run on the SS7 card:

- Transaction Capabilities Application Part (TCAP) for transfer of non-circuit information between signaling points
- Signaling Connection Control Part (SCCP) for transfer of messages other than call setup.

Both TCAP and SCCP can run with any User Part feature or supported variant.

- After the software is enabled, it is available to all of the hardware on the system. For example, if you buy a TCAP license for a single card, the TCAP module will also be available to all other SS7 cards installed in that chassis.

The number of links enabled on the SS7 card is reported in Byte 19 in both the *Card Status Query* and the *Card Status Report* messages.

The licensed spans or links that the card shipped with remain available even if an SS7, T1, or E1 card is removed or if the configuration is lost. However, any license upgrades made to that card are lost. You must re-download the product license from the host to re-activate the software or hardware feature.

Important! If you buy a new SS7 card and the serial number is different from the one you replaced, you need a new product license.

For details, refer to *Downloading License Keys to the CSP* in the *Licensing Overview* chapter in the *Developer's Guide: Overview* and the *Product License Download* message (0x0079) in the *API Reference.*

Customizing

You can customize Dialogic's SS7 environment by using API messages and the PPL for a high-level of flexibility that supports a wide range of protocol and application requirements.

Examples of customization using the API include the following:

- Circuit Identification Code (CIC) Configuration
- Signaling Stack, Link, and Route Configuration
- Default outgoing SS7 Parameters (*PPL Configure*)
- Protocol timers (*PPL Timer Configure*)
- ISUP and TUP message formats
- Dual-seizure control logic
- *Request for Service* message format (*PPL Configure*)

PPL enables you to customize your SS7 environment to specific SS7 variants. You configure SS7 variants on a per-object basis, such as a signaling link, link set, or route, allowing for rapid compliance with international standards.

SS7 Network Architectures

Within an SS7 network, there are three major components:

- Service Switching Points (SSP) - perform call processing on calls that originate, tandem, or terminate at the CSPs.

- Signal Transfer Points (STPs) - relay messages between network CSPs and databases.
- Service Control Points (SCP) - contain centralized network databases, which provide enhanced services.

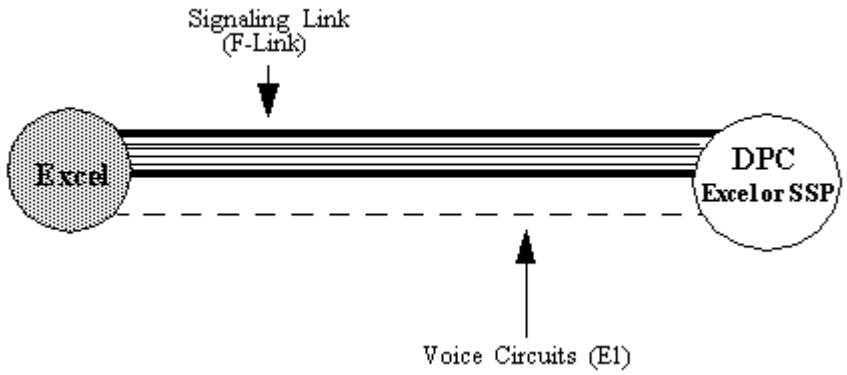
The SS7 cards permit an CSP to act as an SSP in an Advanced Intelligent Network (AIN). The SS7 card supports the following two architectures:

- International Telecommunications Union - Technology Sector (ITU)
- American National Standards Institute (ANSI)

Associated Mode Signaling

Associated Mode Signaling, shown in the next figure, is the typical international (ITU-TS) architecture. Voice circuits between the CSP and the destination point code (DPC) are connected over T1, E1 or J1 carriers. SS7 signaling between the CSP and the DPC is over a signaling link directly connecting the platform to the SP/SSP. There can be multiple signaling links between an CSP and an SP.

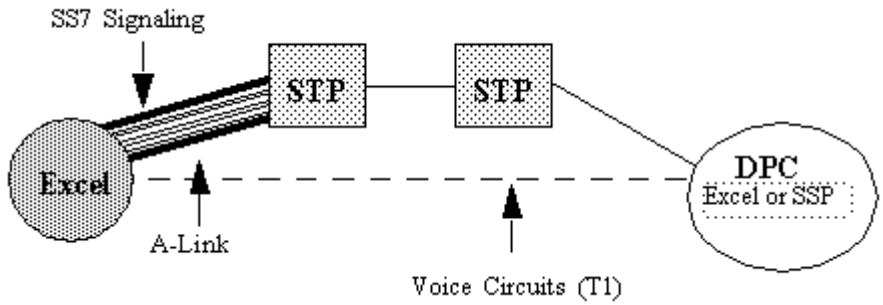
Figure 2-1 Associated Mode Signaling



Quasi-associated Mode Signaling

Quasi-associated Mode Signaling, shown in the next figure, is the typical North American (ANSI) architecture. A T1 carrier connects voice circuits between the CSP and a DPC, typically either another CSP as an SP or an SSP. SS7 signaling between the CSP and SP/SSP is over a signaling link through one or more STPs.

Figure 2-2 Quasi-associated Mode Signaling



SS7 Point Codes

The following discussion of point codes applies to any point codes:

- Originating Point Code (OPC)
- Adjacent Point Code (APC)
- Destination Point Code (DPC)

See also point code information in the *SS7 Route Configure* message in the *API Reference*.

ANSI Point Codes

An ANSI point code is assigned as DPC through Dialogic's *SS7 Route Configure* message (0x5F). Right after the AIB there are four bytes for DPC. The most significant byte (MSB) byte is always 00 unless the DPC is being used to deconfigure a DPC, in which case all bytes are FF. The other three bytes receive the three elements of the point code, with Member portion of the point code in the least significant byte (LSB) position.

Meaning	Network	Cluster	Member
Binary	8 Bits	8 Bits	8 Bits
Hex	FF	FF	FF
Decimal	255	255	255

ANSI Example:

Decimal Point Code = 158-077-099

1. First, convert each part of the point code into hex as follows:
   9E-4D-63

2. Convert each byte to a 32-bit Dialogic format and use this as the last two bytes of the point code in a Dialogic message. Zero-fill the first byte of the point code field:

Dialogic bytes = 0x00 0x9E 0x4D 0x63

ITU Point Codes

An ITU point code is assigned as DPC through Dialogic's *SS7 Route Configure* message (0x5F).

ITU Example:

Decimal Point Code = 2-2-2

1. First, convert each part of the point code into binary, padding it with zeroes to conform to a 3-8-3 bit format (totaling 14 bits) as follows:

   010-00000010-010
   Note that the center part (country) has been padded out with zeros to make up the full 8 bits required for the country part of the point code.
2. Concatenate the whole thing into a single string, as follows:

   01000000010010
   The string should be 14 bits long.
3. Counting from the right, form two groups of bits, the Least Significant Bits and the Most Significant Bits:

   010000 00010010
4. Pad the 6-bit group with zeros to make two full bytes:

   00010000 00010010
5. Convert each byte to hex and use this as the last two bytes of the point code in an Dialogic message. Zero-fill the first two bytes of the point code field:

   Dialogic bytes = 0x00 0x00 0x10 0x12

JT Example:

Decimal Point Code = 2-2-2

1. First, convert each part of the point code into binary, padding it with zeroes to conform to a 7-4-5 bit format (totaling 16 bits) as follows:

   0000010-0010-00010
   Note that the center part (country) has been padded out with zeros to make up the full eight bits required for the country part of the point code.
2. Concatenate the whole thing into a single string, as follows:

   0000010001000010

The string should be 16 bits long.

3. Counting from the right, form two groups of eight bits, the Least Significant Bits and the Most Significant Bits:

   00000100 01000010

4. Convert each byte to hex and use this as the last two bytes of the point code in a Dialogic message. Zero-fill the first two bytes of the point code field:

   Dialogic bytes = 0x00 0x00 0x04 0x42

SS7 Software Architecture

Overview

Dialogic uses the PPL environment to implement the SS7 stack. This feature makes the stack modular and easy to customize. The SS7 software consists of the following modules:

- L3P
- ISUP
- L3P TUP
- TUP
- TCAP
- SCCP
- MTP

Each signaling stack must include MTP, L3P, and at least one user part. Selection of a specific module automatically uses the associated default PPL components.

BT IUP

Dialogic implements BT IUP, a variant of L3P TUP. See *TUP, BT IUP & SSUTR2 (5-1)*, for an example configuration file and call flows for the BT IUP configuration. See *SS7 PPL Information* and the SS7 Signaling Stack Configure 0x5C for additional data on BT IUP.

Call Control

Call control, circuit supervision, and maintenance can be managed using either the ISDN User Part (ISUP) or Telephone User Part (TUP) interfaces. The call control user part for a CIC is configured with the *SS7 CIC Configure* message.

SS7 Signaling Stack Integration

The figures on the following pages show how the SS7 Signaling Stack integrates into the CSP architecture. Default protocol component sets for each module are stored as part of system software and do not require downloading. Selection of a specific module automatically uses the associated PPL components.

Figure 2-3 SS7 Software Architecture (Single Stack)

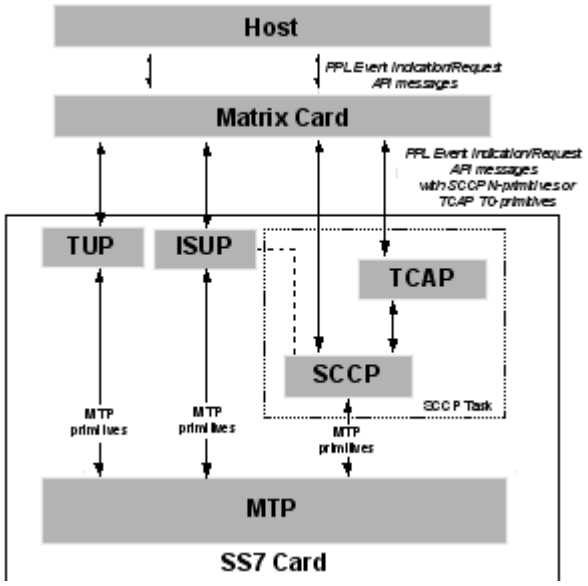
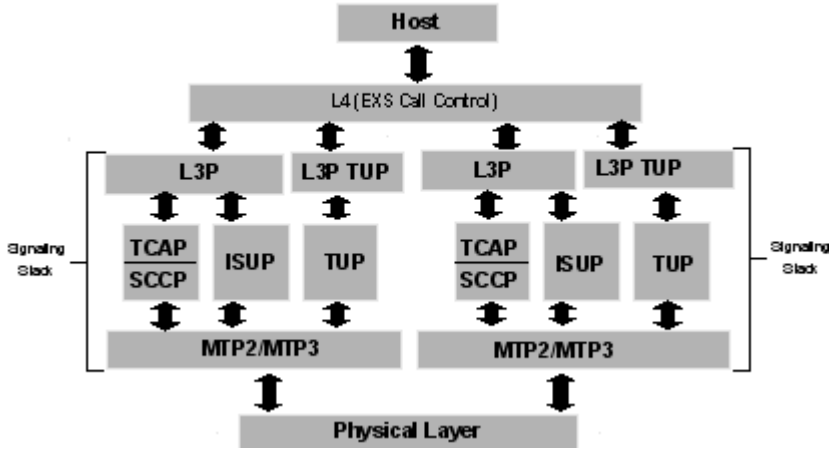


Figure 2-4 SS7 Software Architecture (Multiple Stacks)



Layer 3 Plus (L3P) L3P is an interface between Layer 3 Call Control (ISUP/TUP) and Layer 4 Call Control on the Matrix Controller, and Layer 5 (the host application). It also manages application-specific variants for call control.

L3P includes the following PPL Components:

- L3P CIC Control

Formats the presentation of SS7 data to and from the host and manages call control.

- L3P Link Management

Manages signaling links and formats presentations of SS7 signaling link control to and from the host.

- L3P Telephone User Part (TUP)

L3P TUP is an interface between Layer 3 (TUP), Layer 4 Call Control on the Matrix Controller, and Layer 5 (the host). L3P TUP formats information from Layer 3 into the programmed format for the host and host formatted information for Layer 3. It also manages application-specific variants for call control.

ISUP Integrated Services Digital Network User Part (ISUP) is the protocol for voice and non-voice services. ISUP includes the following PPL components:

- ISUP CPC (Call Processing Control)

CPC manages the call processing portion of ISUP. It manages both incoming and outgoing call setup and tear down.

- ISUP SPRC

SPRC acts as an ISUP message router. It validates and routes incoming ISUP messages to the correct ISUP state machine. It also sends outgoing messages to MTP.

- ISUP MPC/CSC (Maintenance Process Control/Circuit Supervision Control)

A collection of state machines that manage circuit maintenance, such as group blocking/unblocking and circuit reset.

Standard	Component
ANSI	BLR, BLS, CCI, CRI, CGRR, CGRS, CRR, CRS, GBUR, GBUS, UCIC
ITU-TS	BLR, BLS, CCI, CGRR, CGRS, CRCR, CRR, CRS, HGBR, HGBS, HLB, HRB, MGBR, MGBS

TUP Telephone User Part (TUP) includes the following components:

- TUP CPC

CPC manages incoming and outgoing call setup and tear down.

- TUP SPRC

SPRC validates and routes incoming TUP messages to the correct TUP state machine. It also sends outgoing messages to MTP.

MTP

Message Transfer Part (MTP) is responsible for end-to-end reliable delivery of messages generated by its local user part (for example, ISUP). The MTP module comprises the MTP2 and MTP3 layers. MTP includes protocols and procedures to handle network failures and to dynamically reconfigure signaling resources to prevent message loss or duplication and out-of-sequence delivery.

MTP includes the following PPL components:

AERM	RCAT	TCRC
CC	RSRT	TLAC
HMDT	RTAC	TPRC
HMRT	RTCC	TRCC
IAC	RTPC	TSFC
LLSC	RTRC	TSRC
LSAC	SLTC	TXC
LSC	TCBC	
RC SUERM	TCOC	

SCCP/TCAP

Signaling Connection Control Part (SCCP) provides both connectionless and connection-oriented network services above MTP Level 3. Transaction Capabilities Application Part (TCAP) provides transaction and remote operation capabilities to a large variety of applications distributed over the CSP and service centers in the network. Dialogic's SCCP/TCAP development supports the following connectionless services:

- Specialized Routing
- Data Transfer
- Management Control

SCCP/TCAP Components

The following list contains the SS7 SCCP PPL components and their IDs:

0x65 SCLC - SCCP Connectionless Control

0x66 SCRC - SCCP Routing Control

0x67 SUSI - SCCP User Interface

0x68 SPPC - Signaling Point Prohibited Control

0x69 SPAC - Signaling Point Allowed Control

0x6A SPCC - Signaling Point Congestion Control

0x6B SSPC - Subsystem Prohibited Control

0x6C SSAC - Subsystem Allowed Control

0x6D SSTC - Subsystem Status Test Control

0x6E BCST - Broadcast

0x6F LBCS - Local Broadcast

The following list contains the SS7 TCAP PPL components and their IDs:

0x70 TUSI - TCAP User Interface

0x71 CCO - Component Portion Control

0x72 ISM - Component Portion

0x73 TCO - Transaction Coordinator

0x74 TSM - Transaction State Machine

Configuring SS7

Purpose The table below is a summary of basic SS7 configurations you must perform in addition to the other CSP configurations. A detailed configuration sequence follows this summary.

Before you begin Acquire the Originating Point Code (OPC) and Adjacent Point Code (APC) before configuring the SS7 stack.

Configuring SS7 Use the following configuration sequence to bring SS7 signaling links and voice circuits in service. The steps below are used for an SS7 PQ card or SS7 Series 3 card. More details are provided about each step following the table.

Step	Action	API Message(s)
1	Assign and configure spans	Assign Logical Span ID T1/E1 Span Configure
2	Configure SS7 license	Product License Download
3	Configure SS7 cards	CCS Redundancy Configure
4	Configure Signaling Stacks	SS7 Signaling Stack Configure
5	Configure Signaling Link Sets	SS7 Signaling Link Set Configure
6	Configure Signaling Links	SS7 Signaling Link Configure
7	Configure Signaling Route(s)	SS7 Signaling Route Configure
8	Assign Voice Circuits (CICs)	SS7 CIC Configure
9	The following entities must be brought in service in this order and messages must sent separately for each: 1. Spans 2. Signaling links 3. CICs	Service State Configure (This message must be sent for each entity specifying the correct AIB.)

Assign and Configure Spans Assign Logical Span IDs to spans on T1, E1, or J1 line cards you want to use for SS7 with the *Assign Logical Span ID* message.

Set the span format to Clear Channel by using the *T1 Span Configure* or *E1 Span Configure* messages. This step prevents line cards from attempting to extract channel associated signaling from the spans.

Configure SS7 Cards

Configure SS7 cards in the system as primary or secondary using the *CCS Redundancy Configure* message. Independent cards are assigned as a primary card. In a redundant pair of cards, one is assigned as the primary card and the other as the secondary.

Subsequent SS7 configuration messages are sent only to primary SS7 cards, which forward the configuration to the secondary card if there is one.

- When configured as a pair, SS7 cards operate in a primary/secondary mode with respect to the call control user part. Signaling links operate in a load-sharing mode.
- Configuration information can be retrieved with the *CCS Redundancy Query* message.

See *Configuring SS7 Card Redundancy (2-30)* for more information.

Example using CCS Redundancy Configure

The following example assigns a single SS7 card as a primary card. See the *CCS Redundancy 0x005B* message for more details.

```
Trace: H->X FE 00 0D 00 5B 00 SN NI 00 02 01 01 SA 01 01
   SB 00 CS
SN = Sequence Number
NI = Node ID
SA = Slot of first SS7 Card
SB = Slot of second SS7 Card (If there is no second card,
   this value should be 0xff)
CS = Checksum
```

Configure Signaling Stacks

Configuring multiple signaling stacks allows the CSP to interface with multiple SS7 networks. Use the *SS7 Signaling Stack Configure* message to create a stack consisting of three layers: MTP, Call Control User Part (ISUP or TUP), and L3P. If using TUP, you must also configure L3P TUP. Each layer is assigned a variant (ITU-TS or ANSI).

The CSP supports up to four signaling stacks, identified by a Stack ID ranging from 0x00 to 0x03. There is no limitation on how you can allocate SS7 stacks among SS7 cards. For example, you can configure

multiple signaling stacks on a single SS7 card or you can configure one signaling stack on each SS7 card, up to a maximum of four stacks per CSP or multi-node system.

Signaling stacks are assigned to SS7 objects (CICs, links, link sets, routes) as each object is configured.

Configuration information can be retrieved with the *SS7 Signaling Stack Query* message.

Example using SS7 Signaling Stack Configure

The following example sets the OPC of the CSP to 0x111111 and configures all four modules to ITU. See the *API Reference* for more details on the SS7 Signaling Stack Configure message.

```
Trace: H->X FE 00 16 00 5C 00 SN FF 00 01 21 02 01 00 00
   11 11 11 04  01 01 04 01 03 01 05 01
```

Configure Signaling Link Sets

A signaling link set is an abstract path between the CSP and an APC (Adjacent Point Code), to which physical signaling links are added. The same signaling stack must be assigned to all links in a link set. A typical configuration consists of two link sets.

Signaling links in a link set are “load sharing,” that is, signaling traffic is distributed equally on the links to optimize efficiency.

When you configure multiple stacks on an SS7 card, the Link Set IDs assigned to each signaling stack must be different. For example, if Link Set ID is assigned to Stack 1, it cannot also be assigned to Stack 2.

However, if you configure multiple signaling stacks on different SS7 cards, the Link Set IDs assigned to the signaling stacks are independent. For example, Link Set ID 1 can be assigned to Stack 1 on one card and Stack 2 on another card.

To assign Link Set IDs, use the *SS7 Signaling Link Set Configure* API message. Use the *SS7 Signaling Link Set Query* message to retrieve configuration information.

Example using Signaling Link Set Configure

The following example uses the *SS7 Signaling Link Set Configure* message to define Link Set 0 going to APC 0x00 0x22 0x33 0x44.

```
Trace: H->X FE 00 0F 00 5D 00 SN FF 00 01 1E 02 00 00 00
   22 33 44
```

Configure Signaling Links

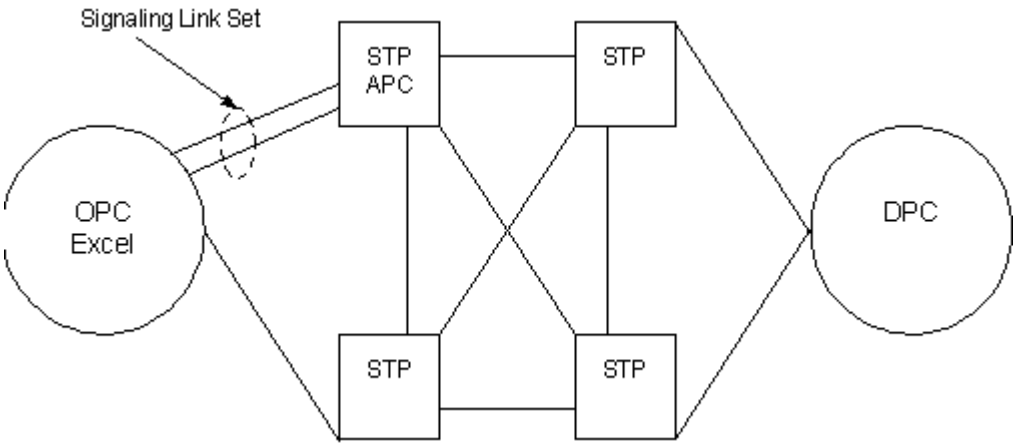
A signaling link is a point-to-point connection between two SS7 point codes (in this case, an CSP and an STP). The *SS7 Signaling Link Configure* message assigns a physical location in the CSP (timeslot) and a previously configured Signaling Link Set to a signaling link.

Each SS7 card supports either 2, 4, 8, or 16 signaling links depending on which model you purchase. For redundant card pairs, Link IDs 0x00–0x0F must always be assigned to the primary card; Link IDs 0x10–0x1F must always be assigned to the secondary card.

When you configure multiple stacks on an SS7 card, the Link IDs assigned to each signaling stack must be different. For example, if Link ID is assigned to Stack 1, it cannot also be assigned to Stack 2.

However, if you configure multiple signaling stacks on different SS7 cards, the Link IDs assigned to the signaling stacks are independent. For example, Link ID 1 can be assigned to Stack 1 on one card and Stack 2 on another card.

Figure 2-5 Configure Signaling Links



Example using SS7 SIgnaling Link Configure

The following example of the *SS7 Signaling Link Configure* defines span 1/channel 23 as a signaling link on Link Set 0, using a data rate of 64 Kbps.

```
Trace: H->X FE 00 14 00 5E 00 SN FF 00 02 1F 03 00 00 00
   0D 03 00 01 00 00 00 00
```

Configure Signaling Routes

A signaling route defines a path between a signaling stack and a DPC. Each SS7 card supports up to 255 routes, identified by a Route ID ranging from 0x00 to 0xFE. Use the *SS7 Signaling Route Configure* message to assign the Route IDs.

The *SS7 Signaling Route Configure* message contains a *Destination ID* field, which defines the relationship of a signaling stack and a specific DPC. Each SS7 card supports up to 64 distinct Destination IDs (0x00–0x3F). All routes between a specific signaling stack/DPC pair must use the same Destination ID. Multiple stacks on the same SS7 card must have unique Destination IDs.

When you configure multiple stacks on an SS7 card, the Route IDs assigned to each signaling stack must be different. For example, if Route ID is assigned to Stack 1, it cannot also be assigned to Stack 2.

However, if you configure multiple signaling stacks on different SS7 cards, the Route IDs assigned to the signaling stacks are independent. For example, Route ID 1 can be assigned to Stack 1 on one card and Stack 2 on another card.

Example using SS7 Signaling Route Configure

The following example of the *SS7 Signaling Route Configure* assigns a route to DPC 0x00 0x33 0x44 0x55. See the *API Reference* for the format of *SS7 Signaling Route Configure*.

```
Trace: H->X FE 00 14 00 5F 00 SN FF 00 01 20 05 00 00 00
   00 01 00 33 44 55 00 09
```

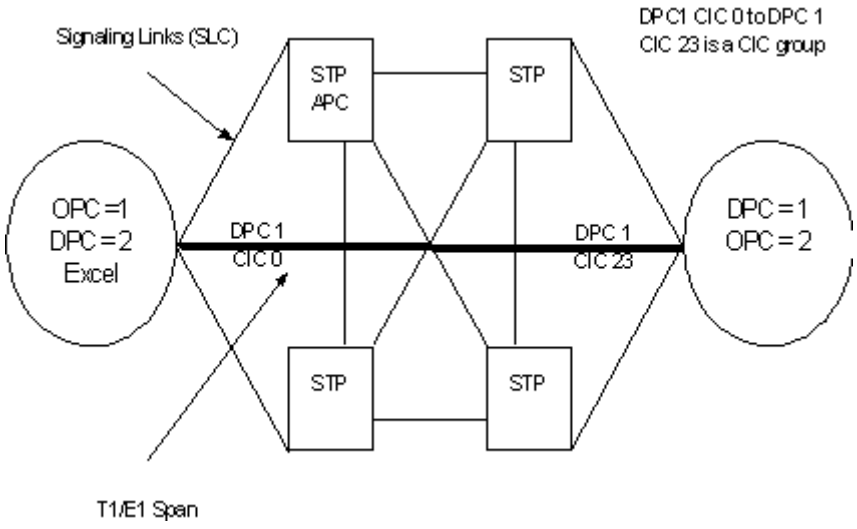
Assign CICs

Voice circuits controlled by SS7 Signaling Links are identified by a Circuit Identification Code (CIC). A specific voice circuit is identified in the network by its unique DPC-CIC combination. The *SS7 CIC Configure* message assigns a DPC/CIC code pair and a User Part (ISUP or TUP) to physical voice circuits. See *Configuring Virtual CIC Format (5-31)* for virtual configuration.

- CICs can be assigned to any timeslot in the system.
- CICs in a CIC group must be on the same span.

- CICs and signaling links can reside on the same span.
- Use the *SS7 CIC Query* message to retrieve configuration information.

Figure 2-6 Assign CICs



Example using SS7 CIC Configure

The following example configures a CIC Group on a T1 span. CIC Group 1 is assigned to span 0, with channel 0 assigned as the base CIC (CIC 0). The remaining channels on the span (1–23) are assigned CIC values 1–23.

```
Trace: H->X FE 00 15 00 6A 00 SN FF 00 01 14 07 00 01 00
   00 18 00 01 02 03 01
```

Adding CIC Information

This feature is used to add Circuit Identification Code (CIC) information in the *Request for Service with Data* and *Outseize Control* messages. The SS7 Address Information data ICB 0x66 defines this information. To enable this feature, on a per channel basis, configuration byte 0x2E of PPL component L3P CIC (0x000F) must be set to 0x01using the *PPL Configure* message.

Once this feature is enabled, ICB 0x66 provides the addressing information in the *Outseize Control* message depending on a positive or negative acknowledgement from the Status (MSB, LSB) field. If the Status field indicates a positive acknowledgement (ACK) 0x10, the ICB data is added. Note that the State field is not present. If the Status

field indicates a negative acknowledgement (NACK), the ICB data is added after the State field. The State field indicates the state of the call processing channel when a NACK is received.

This feature is backward compatible with earlier releases.

Bring Spans, CICs, and Signaling Links in Service

When all the configuration is complete, use the *Service State Configure* message to establish a connection with the network and to begin call processing. Upon successful completion, the CSP sends the host a *DS0 Status Change* message with a status of in service.

SS7 CIC Group Querying

Overview

This host initiated feature enables the customer to determine what CIC groups are configured in order to perform detailed queries, for each CIC group, without any prior knowledge of the existing configuration. The feature supports the *SS7 CIC Query* (0x0067) message in order to display CIC group information:

SS7 CIC Query (0x0067) Message

This host initiated API message, with the following parameters, will be used to query all of the CIC groups for a stack.

- SS7 Stack (0x08) AIB with SS7 Stack ID
- Destination Point Code (DPC) (0xFFFFFFFF) is used to query all CIC groups for the stack. The resultant CIC group information can be used to query a particular CIC group.

From the host perspective, the AIB in the response message is used to differentiate between the following queries:

- For existing CIC group queries: to query information on a particular CIC group, use the SS7 CIC AIB (0x14).
- For CIC group queries: to query information on all of the configured CIC groups, use SS7 Stack AIB (0x08).

SS7 CIC Traffic Over the DS3 Line Card

Overview

The DS3 line cards support SS7 CIC traffic. Under normal operation, a DS3 line card can handle 28 spans, and an CSP chassis can support up to three DS3 line cards. If you wish to support SS7 CIC traffic on all three line cards, you must reconfigure one of the DS3 line cards to 24 spans using DIP Switch 4 and 6. The table below shows what the DIP switch settings are.

DIP Switch	Setting	Spans usable for CIC
Switches 4 and 6	Both ON	All physical spans can be used for CICs.
Switches 4 and 6	Both OFF	Only physical spans 0-23 can be used for CICs.

Causes of SS7 Signaling Link Problems

Purpose

This section describes common causes that prevent SS7 signaling links from coming into service or alignment.

Point Code Mismatch

The OPC (Originating Point Code) as defined in the *SS7 Signaling Stack Configure* message must match the value that the distant end signaling point expects. Also, the distant end's point code must match the APC (Adjacent Point Code) and DPC (Destination Point Code) values defined in the *SS7 Signaling Link Set Configure* message and *SS7 Signaling Route Configure* message.

Signaling Link Code Mismatch

The Signaling Link Code (SLC) is a number (0-15) which is assigned by both ends to identify a specific link within a link set. The SLC defined in the *SS7 Signaling Link Configure* message must match the SLC value assigned to the link by the distant end.

Network Indicator Mismatch

The Network Indicator (NI) value is defined by two bits (therefore values 0-3 are possible). The default value of the Network Indicator is set to National (0x02) for both ANSI and ITU. Some networks may require the Network Indicator to be set to International (0x00) or one of the spare values (0x01 or 0x03). To change the NI value, PPL Configure Messages need to be sent to components MTP3 HMRT and MTP3 HMDT. See the *API Reference* for complete information on configuration byte locations and values.

Link Status Signaling Unit Size Mismatch

By default, the CSP transmit a Link Status Signaling Unit (LSSU) with a 1-octet status field. Some signaling points may require a LSSU size of 2-octet. To change the LSSU size, a PPL Configure Message needs to be sent to component MTP2 TXC. Consult the *API Reference* for complete information on configuration byte locations and values.

Path and Rate Problem

The span carrying the signaling link must be defined for clear channel operation, in service, and not experiencing slips. Both parties must agree upon the timeslot used to carry the signaling link. Additionally, the data rate of the signaling link must be the same on both sides. The *Service State Configure* message for signaling links is different than the

Service State Configure message used for channels. The signaling link timeslot and data rate are defined in the *SS7 Signaling Link Configure* message.

No Route Defined

A valid SS7 route must be defined to the destination to enable the CSP to send messages. This is typically indicated by received Signaling Link Testing Message (SLTM) with no SLTMs being sent by the CSP.

TCP/IP with SS7

Overview

Transmission Control Protocol/Internet Protocol (TCP/IP) provides communication between an SS7 card and a host. The SS7 card, enabled with a local host, communicates directly with the host by using Ethernet technology. The local host uses TCP/IP over a direct Ethernet connection to communicate with the host. An Ethernet connection for each SS7 card is required and each SS7 card must be on the same network as the Matrix Controllers. The communication scheme for a local host is similar to the way the Matrix Controller communicates with a host. Physically, the local host can be on the same computer as the Matrix Controller host or it can be on a separate host computer, depending on the user's hardware setup.

SS7 TCAP/SCCP Local Host Connection

The direct connection between an SS7 local host and a host is configured by using the *SS7 SCCP/TCAP Configure* (0x77) message. Once configured, SCCP/TCAP traffic is routed to and from the SS7 local host. Only the *PPL Event Request*, *PPL Event Indication,* and *Poll* messages are sent over this connection. All other messages must go through the standard host-Matrix Controller connection or they will be rejected.

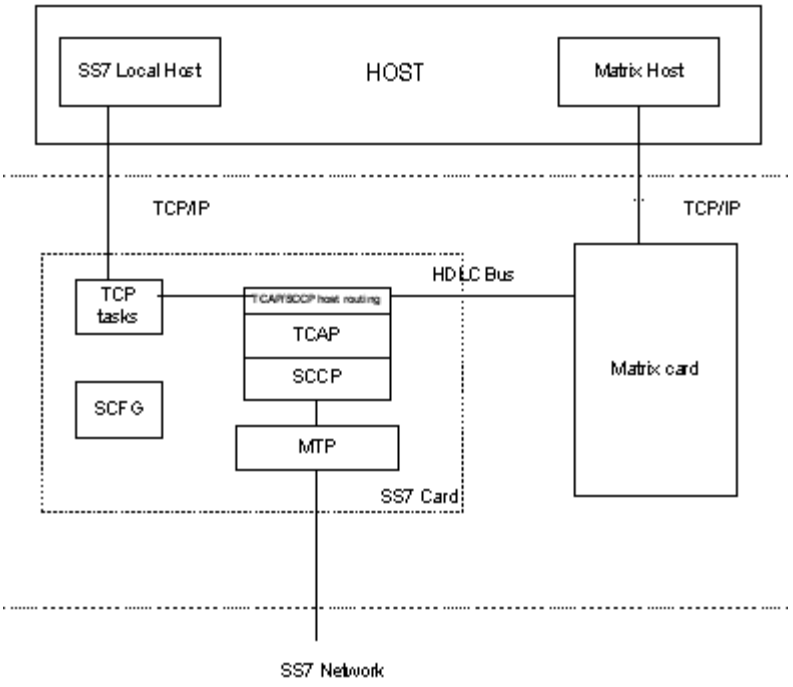
Each SCCP/TCAP stack/subsystem can be configured to use an SS7 local host using the *SS7 SCCP/TCAP Configure* message. The default host for each stack/subsystem is the primary Matrix Controller. Whether you use a SS7 local host or Matrix Controller host, configuration is performed on a per stack/subsystem basis. You can query the current host configuration using the *SS7 SCCP/TCAP Query* (0x78) message.

A Matrix Controller card switchover or reset will not impact TCAP transactions over the SS7-host connection.

Architecture Overview

A TCAP direct host connection is implemented by adding a thin layer on top of the current TCAP message routing function. TCAP/SCCP software is modified to keep track of the host information when receiving a PPL event request from the host. Configuration and query options are provided for the TCAP direct host routing. Whether you use a SS7 local host or Matrix Controller host, configuration is performed on a per stack/subsystem basis. See figure below for the architecture overview.

Figure 2-7 Architecture Overview



Redundancy

The SS7 TCAP/SCCP Local Host Connection is supported with redundant SS7 cards. When an SS7 card switchover occurs, it is possible that some messages will get lost. Always send or receive the messages to the active card only when cards are in the active or standby mode. When the active card resets, the SS7 card switchover starts. Then, the host can communicate with the standby card directly. See *Configuring SS7 Card Redundancy (2-30)* in the Configuration section.

TCP/IP Performance

Transmission Control Protocol (TCP) protocol messaging is used to optimize response time to and from the local host. A message must first be received from the host so that the port number can be stored for use in subsequent indications. Dialogic's SS7 card design allows for a maximum of 65,536 sequence numbers on the card. The sequence numbers indirectly have an impact on the retry mechanism for CSP-based (service card) messages. The retry mechanism allows the service card to resend any unacknowledged messages to the host on a regular interval for a defined number of retry times. With the use of the direct TCP/IP link between the service card and the host, a high volume of

messages can be processed in a given time period, which in turn will often result in a large number of unacknowledged messages outstanding. Two-hundred CSP-based messages can be saved.

Polling

A *Poll* (0xAB) message is sent to the host from each SS7 card every two seconds or whenever there is an SS7 card state change. The *Poll Interval Configure* and *Poll Request* messages, as well as the Host Link Failure Detection feature, are not supported in this implementation of the SS7 card-to-host polling.

TCAP Dialog Timeout

The TCAP dialog timeout feature allows out-of-date dialogs to be aborted when the dialog does not get terminated correctly and the dialog ID is hanging. This may occur, for example, when messages get dropped during an SS7 card switchover. When a dialog timeout occurs, TCAP will try to abort the dialog using the abort reason defined in component, TCAP TUSI (TCAP User Interface - 0x70), using Config Byte 0x07. It will also send a new TCAP PPL event “Dialog timeout abort” to the host using PPL Event ID, 0x20, in the TCAP TUSI component. The TCAP dialog timeout timer is measured as “how long the dialog is idle.” In other words, how long there are no messages exchanged for a dialog.

The TCAP dialog timeout feature is enabled with Config Byte 0x06 in component TCAP TUSI. The dialog timeout threshold timer is also configurable using the PPL Timer, 0x01, in component, TCAP DHA (TCAP Dialog Handling - 0x75).

More Information

The PPL information for both ANSI and ITU standards related to this feature is provided in *SS7 PPL Information.*

Configuring TCP/IP with SS7

Purpose This section describes how to set up communication between a host and an SS7 card using TCP/IP. Redundancy and example configuration traces are also provided.

Before you begin The host needs to support the TCP/IP protocol. The host communication to the directly-connected SS7 card is similar to that of the Matrix Controller card. The host selects to communicate with the Matrix Controller card or the SS7 card by using a different IP address.

IP Addressing TCAP messaging can be sent to the Matrix Controller or SS7 Ethernet port. For information on configuring the IP address on the Ethernet port see Host Communication Module in the *API Developer's Guide: Overview.*

Configuration Follow the steps below to create communication between an SS7 card and a host.

Step	Action
1	Establish an Ethernet connection to the Matrix Controller card.
2	Configure the IP addresses. After the system software is successfully loaded to an SS7 card the first time, you must set the IP address and subnet mask of each SS7 card using *IP Address Configure* or BOOTP server. The cards retain the IP address and subnet mask even if the system software is reloaded or the system is powered down. You can use the *IP Address Query* message to get the IP address of the SS7 card after it is set. If an SS7 card does not have an IP address set, it attempts to use BOOTP and RARP to find a BOOTP or RARP server which can supply the IP address. If these fail, the card comes up with no IP address, which disables the use of the "local host" connection to the SS7 card.
3	Configure the host connection Using Configuration Type 0x08 in the *SS7 SCCP/TCAP Configure* message, delete Matrix Controller host connection. Using Configuration Type 0x08 in the *SS7 SCCP/TCAP Configure* message, add a local host. For SCCP/TCAP applications, you may configure the SS7 local host for each stack/subsystem.
4	Configure redundant SS7 cards (if applicable) See below.

Redundancy Considerations

An SS7 local host should always communicate with the active SS7 card. The messages directly sent to the standby SS7 card will not be processed. You may set up a socket connection from the host to the standby card, but the only messages that go through this connection should be the *Poll* messages from the SS7 card to the host. In the case of an SS7 card switchover, it is possible that some messages are lost.

Prior to a switchover, you can configure the TCAP User Interface component (0x70) to either “Reset all the TCAP dialogs” or “Keep the TCAP dialogs” based on the nature of the applications. If the lifetime of the TCAP dialogs is long, for example, the dialogs last for a whole call period, the percentage of dialogs being affected during a switchover will be low. The application may choose to keep all of the dialogs after the switchover. If the application keeps all the TCAP dialogs, TCAP and TCAP applications rely on the TCAP dialog timeout feature to eliminate the aged TCAP dialogs.

TCAP application level error handling procedures are used to recover the errors caused by losing messages for the TCAP dialogs. If the lifetime of the TCAP dialogs is short, for example, the dialog is ended after the call is set up, the percentage of affected dialogs could be high. The application may choose to abort all the TCAP dialogs after switchover. In this case, TCAP will automatically initiate a TCAP restart and all the active TCAP dialogs will be aborted after the previous standby card becomes active. The host can then communicate with the new active card directly afterward for the new dialogs.

Example Configuration Traces

The following are examples of various SS7 Host Connection message traces.

Deleting Matrix Controller Host Connection

The following trace of an *SS7 SCCP/TCAP Configure* message deletes the Matrix Controller host for Stack 0 subsystem 5 so that messages will not be routed to the Matrix Controller host:

```
00  0f   00  77  00  00  ff   00  01  08  01  00  08  05
   02   ff   00
```

Configuring an SS7 Card-Host Connection

The following trace of an *SS7 SCCP/TCAP Configure* message adds a local host for Stack 0 subsystem 5 so that messages will be routed to the SS7 local host:

```
00  0f   00  77  00  00  ff   00  01  08  01  00  08  05
   01  fe  00
```

Querying the Host Connection

The following trace of an *SS7 SCCP/TCAP Query* message queries the local host for Stack 0 subsystem 5:

```
00 0c 00 78 00 00  ff  00 01 08 01 01 01 05
```

> **Important!** These configuration messages are sent to the Matrix Controller card IP address. After the configuration, you may start the TCAP traffic by sending the *PPL Event Request* to the SS7 card IP address. The CSP will send TCAP PPL event indications to the SS7 local host.

Clear All Dialogs

The following trace of a *PPL Event Request* message initiates a “TCAP restart” event to clear all dialogs for Stack 0 subsystem 5:

```
00 11  00  44   00   00   ff    00  01  2a  03   00  05 00
   00 70  00  1e  00
```

Configuring SS7 Card Redundancy

Purpose

Dialogic provides complete redundancy by enabling all SS7 interfaces on an SS7 card to be replicated on a standby card when I/O cards are used with two SS7 cards. The I/O card(s) in conjunction with a pair of SS7 cards, prevents the SS7 card from being a single point of failure. A mirror image of the primary card's signaling configuration is copied to the secondary card. When signaling links are distributed as recommended, if either SS7 card experiences a fault condition, the other card manages all call processing without losing any active calls.

You configure redundancy by using the *CCS Redundancy Configure* (0x5B) message to designate one SS7 card as primary (active) and the other as secondary (standby) prior to configuring the SS7 interface or after completion of configuration.

Dialogic recommends that you send the *CCS Redundancy Configure* message after resetting the configuration on the whole system. The synchronization of databases on the cards involves intensive processing, so you should send the message when there is little or no call activity on the system. After you send the *CCS Redundancy Configure* message, all subsequent configuration is distributed to both cards. For example, during an SS7 card switchover, the stable calls remain active and the other calls are purged with the switchover purge code 0x18.

Before You Begin

It is assumed that two SS7 cards and associated I/O card(s) are properly installed. (See the CSP *Hardware Installation and Maintenance Guide*). Upon power-up, the SS7 cards are automatically brought in service as independent cards.

CAUTION

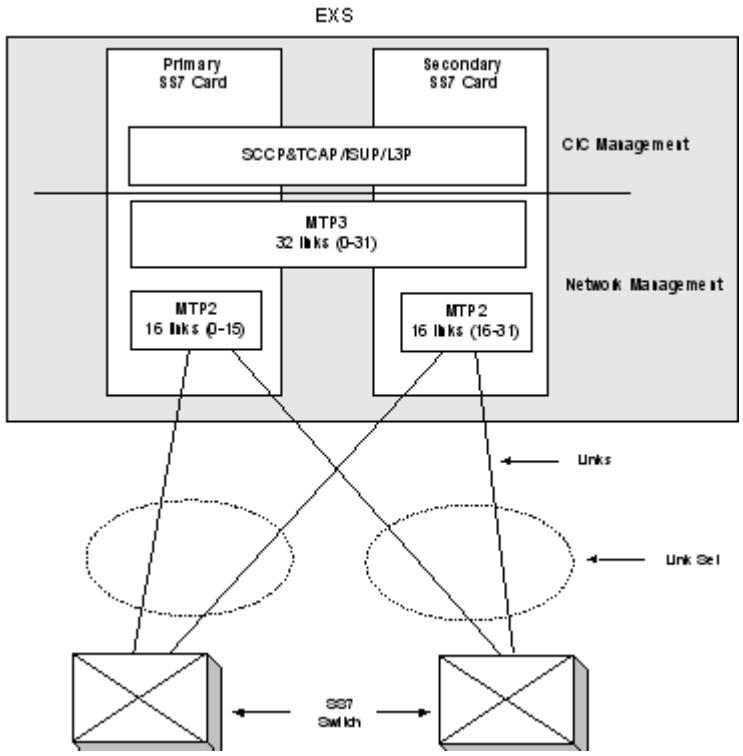
*Always press the **STOP** button on the SS7 PQ or SS7 Series 3 line card in a redundant card pair configuration before removing or inserting the corresponding I/O card. For example, in a redundant SS7 PQ or SS7 Series 3 card pair configuration, be aware that the redundant line card may reset when removing*

or inserting the primary I/O card. If this occurs, the system will lose calls.

Redundancy Architecture

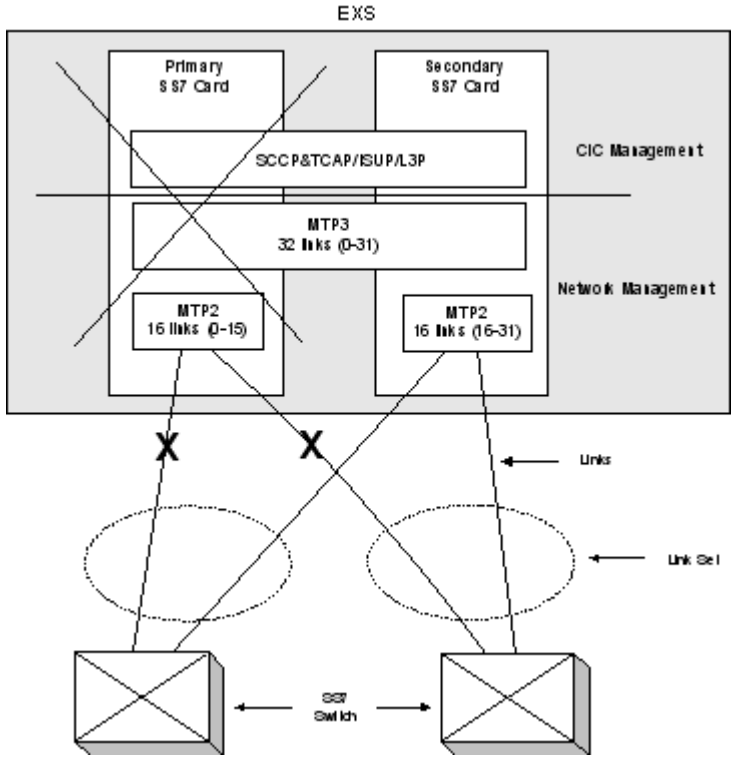
The figure below shows a block diagram of the redundancy architecture and signaling distribution.

Figure 2-8 SS7 Redundancy - Normal Operation



If one of the SS7 cards is reset or faults, the signaling relations between the CSP and remote SS7 CSPs are maintained, as shown in the figure below.

Figure 2-9 SS7 Redundancy - Fault Condition



Configuration

Configure redundancy by designating one SS7 card as the primary card and the other as the secondary card with the *CCS Redundancy Configure* message. You will receive *CCS Redundancy Report* messages indicating the status of each card.

Configuring SS7 cards:

Configure signaling stacks and routes on the primary SS7 card. This configuration is automatically transferred to the secondary card during the synchronization process.

The maximum configuration figures for a single card apply to a card pair as well, except for the number of signaling links, which is doubled. Two 16-link SS7 cards support 32 links as a pair.

When the system configures the signaling links, it automatically assigns Link IDs 0–15 to the primary card and Link IDs 16–31 to the secondary card. As long as redundancy is enabled, these Link IDs must be maintained on the respective cards.

To achieve fully fault-tolerant signaling relations, signaling links within a link set should be distributed across the two SS7 cards. In the case of single-link link sets, the signaling routes should be distributed across the SS7 cards.

Constraint

The *CCS Redundancy Configure* message will fail if the SS7 card in the secondary slot already has any links numbered 0-15 assigned to it. A status of 0xF2 “CCS Redundancy: Not the Primary Slot” returns. Only the SS7 card in the primary slot may have links numbered 0-15 assigned prior to the *CCS Redundancy Configure* message being sent. The host application must keep track of which SS7 cards currently have links assigned.

Redundancy States

A pair of SS7 cards is in either a synchronization state or a stable state while redundancy is enabled:

Synchronization States

The synchronization states are:

0x01	Primary Active, Secondary Synchronizing
0x02	Secondary Active, Primary Synchronizing

Stable States

The stable states are:

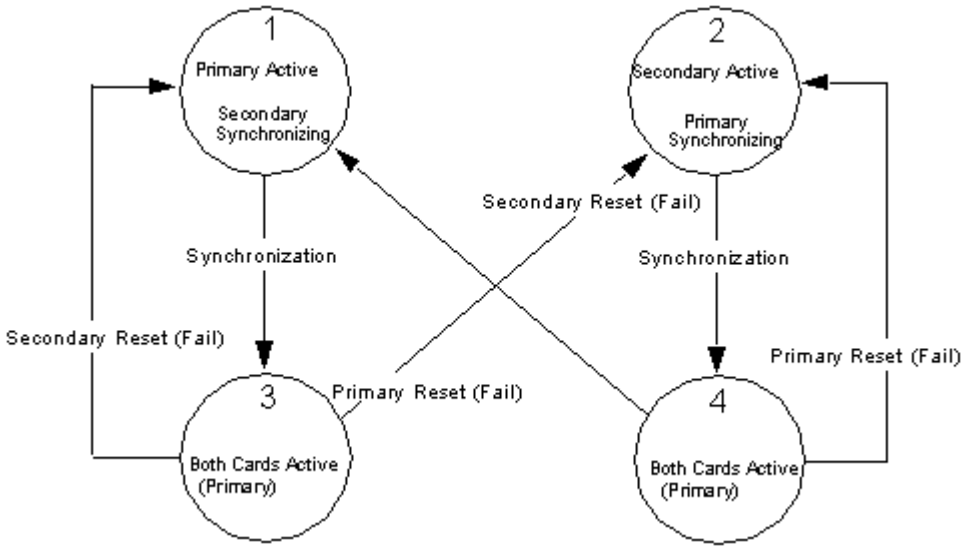
0x03	Both Cards Active (Primary)
0x04	Both Cards Active (Primary)

Important! The distinction between States 3 and 4 is transparent to the host. It is for Dialogic use only. The SS7 card pair should be in either State 3 or 4 during normal operation. In these stable states, both SS7 cards are active and processing calls.

State Transitions

The next figure shows the four states of redundancy and the transition between states due to the resetting of either SS7 card. The host is sent *CCS Redundancy Report* messages for each of the state transitions shown.

Figure 2-10 Redundancy States



SS7 Local Host Initiated Redundancy Switchover

Purpose

The SS7 Local Host Initiated Redundancy Switchover feature allows a host directly connected to the SS7 active and standby cards, by a TCP/IP Ethernet connection, to initiate a SS7 redundancy switchover.

Description

This feature is an enhancement to the existing SS7 Card Direct Host Ethernet Connection feature. Initiating a SS7 card switchover requires that the SS7 local host is connected to both SS7 cards via a direct TCP/IP Ethernet connection.

When the SS7 local host detects an active SS7 card failure, it initiates a SS7 switchover by sending a *Become Active* (0x00A1) API message to the standby SS7 card via the direct Ethernet connection. This causes the active SS7 card to reset (eventually becomes standby) and the standby SS7 card to become active. The host then communicates with the newly active card (previously the standby card) for traffic messages.

The SS7 Local Host Initiated Redundancy Switchover feature is also consistent with the CSP 2000 Matrix Card host interface and switchover scheme.

Switchover Sequence

The SS7 local host is aware of each of the SS7 cards active/standby status based on the *Poll* (0x00AB) API message sent from each of the redundant SS7 cards. The switchover sequence and related figure below describe and illustrate the process of detecting the failure of the active SS7 card through the switchover of the standby SS7 card to active status. Note that the circled numbers on the figure correspond to the steps below.

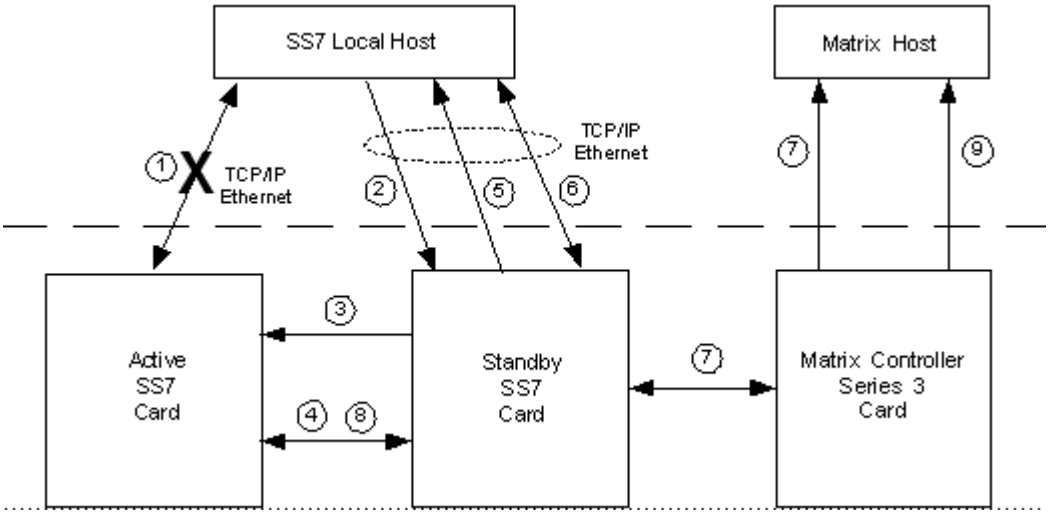
1. When the redundant SS7 cards (active and standby) process messages to and from the SS7 local host, the host may determine that the connection to the active SS7 card failed.
2. The SS7 local host then sends a *Become Active* (0x00A1) API message to standby SS7 card.
3. When the standby SS7 card receives a *Become Active* API message, it sends an internal RESET_CONFIGURATION message to the active SS7 card. The active SS7 card resets upon receiving the RESET_CONFIGURATION message from the standby SS7 card.
4. The standby SS7 card status becomes active immediately, since it is already in synchronization with the previously active SS7 card.

5. The newly active SS7 card then sends a *Poll* API message to the SS7 local host indicating that it is now active.
6. The SS7 local host resumes traffic by processing messages to and from the standby SS7 card.
7. The CSP Matrix Series 3 Card detects a SS7 card status change; it sends a *CCS Redundancy Report* API message to the Matrix host to report this status change.
8. During this time the active SS7 card is synchronizing with the standby SS7 card. After synchronization the previously active SS7 card becomes standby.
9. The Matrix host is informed of the card status change by the *CCS Redundancy Report* and *Card Status Report* API messages after the switchover is complete.

Important! If an active or single SS7 card receives a *Become Active* API message from the SS7 local host, it sends a positive acknowledgement to the local host and no further action is taken.

Important! If a SS7 card, in the synchronization state, (a state other than active, standby, or single) receives a *Become Active* API message from the SS7 local host, a negative acknowledgement with status 0x14 (Invalid Command) is sent to the local host. No further action is taken.

Figure 2-11 SS7 Redundancy Cards Switchover Sequence



SS7 Fault Conditions & Fault Recovery

Overview

SS7 stack operations are based on at least one link in a link set on each card in the redundant pair. The stack operations are maintained through the following fault conditions:

- An SS7 card is:
  - Removed (host receives *Card Status Report* message).
  - Reset by either the **Reset** button or with the *Reset Configuration* message.
  - Taken out of service either by the host with the *Service State Configure* message or by the CSP.
  - Malfunctions and forces a reset.
- The redundancy link between the two SS7 cards is broken because:
  - The I/O card(s) is removed.
  - The card-to-card link on the bus is lost.

Fault Recovery

Important! The **Stop** button (on a given card) should always be pushed before removing that card from the system. The PPL Event Indications may not be generated if the SS7 card is pulled without first pressing the **Stop** button.

The table below shows the results and host action required in response to each fault scenario.

Event	Result	Host Notification	Action Required
SS7 card removed (Active)	- All answered calls are maintained on the remaining SS7 card. - Redundancy is disabled.	PPL Event Indication of Signaling link failure (MTP3 LSAC) * CCS Redundancy Report (Redundancy Status: 0x20 or 0x21) Alarm (Severity: 0x03, Entity: 0x02, Alarm Number: 0x0E) Card Status Report (Card Type: 0xFF)	1. Replace SS7 card. 2. Reconfigure redundancy. 3. Reconfigure signaling links on the replaced card (*SS7 Signaling Link Configure*). 4. Bring Links in service (Service State Configure)

Event	Result	Host Notification	Action Required
SS7 card removed (Standby)	- All answered calls are maintained on the remaining SS7 card. - Redundancy is disabled.	Same as Above	Same as Above
SS7 card reset (Active)	- All answered calls are maintained on the remaining SS7 card. - The SS7 card that reset comes back in service with SS7 configuration maintained. - The CSP automatically synchronizes the SS7 cards and re-establishes redundancy.	PPL Event Indication of Signaling link failure (MTP3 LSAC) * CCS Redundancy Report (Redundancy Status: 0x29 or 0x30) Alarm (Severity: 0x03, Entity: 0x02 Alarm Number: 0x09) Card Status Report for I/O (Card State: 0x00) Card Status Report for SS7 (Card State: 0x00) CCS Redundancy Report (Redundancy Status: 0x03 or 0x04)	No action is required. To suspend synchronization, see *Suspending Synchronization (2-40)* in this chapter.
SS7 card reset (Standby)	- All answered calls are maintained on the remaining SS7 card. - The SS7 card that reset comes back in service with SS7 configuration maintained. - The CSP automatically synchronizes the SS7 cards and re-establishes redundancy.	Same as Above	Same as Above
SS7 card taken out of service (Active)	- All answered calls are maintained on the remaining SS7 card. - Redundancy is disabled.	CCS Redundancy Report (Redundancy Status: 0x20 or 0x21) Card Status Report (Bit 3 of Card Status Byte is set)	

Event	Result	Host Notification	Action Required
SS7 card taken out of service (Standby)	- All answered calls are maintained on the remaining SS7 card. - Redundancy is disabled.	Same as Above	Same as Above
I/O is removed	- The active SS7 card currently processing calls remains in service and all calls are maintained. - The other SS7 card resets and redundancy is disabled.	CCS Redundancy Report (Redundancy Status: 0x22) Alarm (Severity: 0x03 Entity: 0x02 Alarm Number: 0x0E) Alarm (Severity: 0x03 Entity: 0x02 Alarm Number: 0x09) Card Status Report (Card Type: 0xFF)	1. Replace the I/O card. 2. Reconfigure redundancy. 3. Reconfigure signaling links on the replaced card (SS7 Signaling Link Configure). 4. Bring Links in service (Service State Configure)
Card-to-card link is broken	- The SS7 card in the Active state resets.	CCS Redundancy Report (Redundancy Status: 0x01 or 0x02) Alarm (Severity: 0x03, Entity: 0x02 Alarm Number: 0x09) Card Status Report for I/O (Card State: 0x00) Card Status Report for SS7 (Card State: 0x00) CCS Redundancy Report (Redundancy Status: 0x03 or 0x04)	Check I/O card. It has either faulted or been removed.

SS7 Card Synchronization

Overview

The CSP attempts to synchronize SS7 cards following all fault conditions except the removal of the card. When synchronization is complete, redundancy is re-established and the host receives a *CCS Redundancy Report* message.

Suspending Synchronization

All configuration and query API messages (except *CCS Redundancy Query*) are blocked during the synchronization process. Depending on the specific configuration and the amount of traffic being handled, synchronization of the SS7 cards can take a significant amount of time.

Synchronization can be suspended by the host during high traffic periods to avoid the system slow-down incurred by sending a *CCS Redundancy Configure* message with the *Options* field value 0x01 (Suspend Synchronization).

The SS7 card in the Active state resets and returns to service as an independent card and redundancy is disabled. Only the SS7 card in the Active state continues to process calls. The other card can be reconfigured as an independent card or redundancy during slow traffic periods.

SS7 Card Switchover

Important! In a switchover, if PPL Configuration Byte 0x32 for ISUP SPRC component (0x13) is disabled, the host sends no circuit reset message to the network and no purge message is sent to the host. Disabling causes the host to respond to calls that are in a setup state as though they are in an unknown state.

During an SS7 card switchover, the CSP maintains connections for those channels that were connected prior to the switchover. Therefore, answered calls remain active. Calls in the process of setup are torn down. The CSP informs the host of all in-service channel states using the *DS0 Status Change* (0x42) message which sends switchover purge code 0x18. Purging is done over all the channels which have no active calls (incoming or outgoing). Some of them will be idle and some will be in an intermittent state. Purging will bring those channels into a known idle state. See the *API Reference, DS0 Status Change* (0x42) message, for more information about purge code 0x18.

Along with the purge code sent to the host, a Circuit Reset (RSC) message is sent to the network. When a Release Complete (RLC) message is received by the SS7 card, the channel is brought back into service.

SS7 Redundancy: Disabling, Reconfiguring and Querying

Overview

This procedure describes how to disable, reconfigure and query SS7 redundancy. If you need to alter redundancy on a live system, follow the procedure, *In-Service Upgrades (2-44)*.

Before you begin

Before you disable SS7 redundancy using the *CCS Redundancy Configure* message, you must be prepared to allow both SS7 cards to reset and for all call processing to stop.

Disabling Redundancy

Redundancy can be disabled either automatically by the CSP or by the host with the *CCS Redundancy Configure* message. When redundancy is disabled, one of the following values is sent to the host in the *CCS Redundancy Report* message:

Redundancy Disabled Values

0x20	Primary card is out of service
0x21	Secondary card is out of service
0x22	Redundant I/O card is removed
0x24	Redundancy disabled by the host
0x25	Primary card reset during synchronization
0x26	Secondary card reset during synchronization
0x27	Primary card detected link failure to mate during synchronization state (1 or 2)
0x28	Secondary card detected link failure to mate during synchronization state (1 or 2)
0x29	Primary card detected link failure to mate while in stable state (3 or 4)
0x30	Secondary card detected link failure to mate while in stable state (3 or 4)

CSP Disabling of Redundancy

Redundancy is automatically disabled by the CSP under the following conditions:

- Either SS7 card is removed or taken out of service.
- The I/O card is removed.
- A Matrix Controller switchover occurs while the SS7 cards are synchronizing.

See *SS7 Card Switchover (2-40)*.

Host Disabling of Redundancy

You can disable the redundancy configuration from the host by sending the *CCS Redundancy Configure* message and setting the *Secondary Slot* field to 0xFF. Both SS7 cards reset and come back in service as independent cards with the default configuration. You must resend non-default SS7 configuration to both cards.

To disable redundancy without resetting both SS7 cards, remove one of the cards or take it out of service with the *Service State Configure* message.

WARNING

Disabling redundancy with the CCS Redundancy Configure message resets both SS7 cards and stops all SS7 call processing.

Reconfiguring Redundancy

To activate redundancy after replacing an SS7 card, send the *CCS Redundancy Configure* message using the same primary and secondary assignments for the SS7 cards as when redundancy was originally configured.

You must reconfigure signaling links and PPL configuration (*PPL Assign*, *PPL Configure*, and *PPL Timer Configure* messages) for link-related components. Link components are documented in the *Chapter 7, SS7 PPL Information* of this developer's guide.

Querying Redundancy

You can query the redundancy information by sending the *CCS Redundancy Query* message. The CSP returns the state of the redundant pair and identifies the slots of the Primary, Secondary, and Active cards.

The following values are reported to the host in the *CCS Redundancy Query* message to indicate the redundancy status:

Synchronization States

0x01	Primary Active, Secondary Synchronizing
0x02	Secondary Active, Primary Synchronizing

Stable States

0x03	Both Cards Active (Primary)
0x04	Both Cards Active (Primary)

Redundancy Disabled

0x05	Redundancy is being disabled

In-Service Upgrades

Making a Non-redundant System Redundant

To make a non-redundant system redundant, perform the following steps:

Step	Action
1	Install a second SS7 card or an SS7 card and I/O card pair as shown in the CSP *Hardware Installation and Maintenance Guide*.
2	Wait for system to send the *Card Status Report* message for each inserted card.
3	Configure SS7 redundancy.
4	Wait for synchronization to complete (*Redundancy Status* field value of 0x03 or 0x04 in the *CCS Redundancy Report* message).

Replacing an SS7 Card

To replace an SS7 card, perform the following steps:

Step	Action
1	Press the STOP button on the SS7 card.
2	Remove the appropriate SS7 card and replace it with a functional SS7 card.
3	Wait for system to send the *Card Status Report* message for the inserted card.
4	Configure SS7 redundancy. *Configuring SS7 Card Redundancy (2-30)*

Step	Action
5	Wait for synchronization to complete (*Redundancy Status* field value of 0x03 or 0x04 in the *CCS Redundancy Report* message).

Replacing an I/O Card

To replace an I/O card, perform the following steps:

Step	Action
1	Press the STOP button of the SS7 card in front of the I/O cards.
2	Remove the appropriate I/O card and replace it with a functional I/O card.
3	Wait for system to send two *Card Status Report* messages, one for each of the I/O cards.
4	Configure SS7 redundancy.
5	Wait for synchronization to complete (*Redundancy Status* field value of 0x03 or 0x04 in the *CCS Redundancy Report* message).

SS7 Host Link Failure Detection

Overview

The SS7 host link failure detection and handling feature is based on the existing SS7 local host TCP/Ethernet connection feature. The SS7 host link failure detection and handling feature enables the SS7 card to monitor its TCP/Ethernet connection to its local host by checking the messages received from the local host. If the SS7 card has not received any message from its local host for a certain amount of time, the SS7 card will assume a host link failure has occurred. SCCP/TCAP uses an SS7- to-host Ethernet connection to communicate with a application.

When a single card (or both of a redundant pair of SS7 cards) detects a host link failure, the SS7 card can take the corresponding subsystem out-of-service, abort TCAP transactions, and update the SS7 network with the subsystem out-of-service condition. This will prevent the SCCP/TCAP module from receiving new TCAP transactions from the network when the SS7 card cannot deliver messages to the host application.

The SCCP/TCAP subsystem out-of-service procedure also applies for those subsystems on the matrix host when the matrix host link fails.

This feature is supported for both the SS7 PQ card and SS7 HP Series 3 card.

Poll Messages

Polling has to be enabled to enable host link failure detection. See *SS7 Polling and Host Link Failure (2-55)*.

API Messages Used for this Feature

The following APIs, listed in alphabetical order, support the host link failure detection feature for SS7 cards. The format of only three of the messages have been modified to support the feature; they are: *Poll* (0xAB), *System Configuration* (0xAF), and *System Configuration Query (0xB4)*. For more details, see the *API Reference*.

- Alarm (0xB9)

A minor card alarm type, Host link Error, is used to report the receiving host message timeout for the SS7 card. A major card alarm type Host Link Failure is used to report that the SS7 detects a host link failure after four polls.

If the SS7 card does not receive any message from its local host, a minor alarm is sent to the matrix host. The relevant data is:

Byte	Field Description
8	Severity (0x02 Minor)
9	Alarm Type (0x02 Card Type Alarm)
10	Alarm Number (0x10 Host Link Error Detected)

If the SS7 card detects a host link failure to its local host, a major alarm is sent to the matrix host. The relevant data is:

Byte	Field Description
8	Severity (0x03 Major)
9	Alarm Type (0x02 Card Type Alarm)
10	Alarm Number (0x22 Host Link Failure Detected)

- Alarm Cleared (0xC1)

  If the SS7 card receives a local host message before the host link failure timer expires, an *Alarm Cleared* for the minor alarm is sent to the matrix host. The relevant data is:

Byte	Field Description
8	Severity (0x02 Minor)
9	Alarm Type (0x02 Card Type Alarm)
10	Alarm Number (0x10 Host Link Error Detected)

  If the SS7 card detects that the local host link recovers, an *Alarm Cleared* for the major alarm is sent to the matrix host. The relevant data is:

Byte	Field Description
8	Severity (0x03 Major)
9	Alarm Type (0x02 Card Type Alarm)
10	Alarm Number (0x22 Host Link Failure Detected)

- Poll (0xAB)

  The SS7 card supports sending this message to its local host. The third least significant bit of the Card Status field indicates whether the link is up between the mate SS7 card and the local host, and whether the mate SS7 card is receiving messages from the host. The default value is 1, which specifies that host link failure detection is not enabled.

- Poll Interval Configuration (0x9F)

  The format of this message does not change. The host send this message directly to the SS7 card through the local host. The SS7 card processes this message from the SS7 local host. The poll interval can be user-defined within a range of valid values: 0-25 seconds. The default value is 2 seconds. The SS7 card stops polling if the interval timer is 0.

- Poll Request (0x9E)

  The format of this message does not change. The host sends this message to the SS7 card from the SS7 direct connection. The SS7 card processes this message from the SS7 local host, and initiates polling after receiving this message. ***Note***: The host can re-initiate by default after the SS7 card sends a poll to the local host.

- PPL Event Indication (0x43)

  -The format of this message does not change. An SCCP N-State indication is not sent to the host when a host link failure occurs. Normally, when SCCP takes subsystems out-of-service, an N-State is sent to the host.

  -A TCAP TC-U-Abort indication is reported in the *PPL Event Indication* message. Typically when a TCAP restart is initiated, the host will receive a TC-U-Abort for the aborted TCAP transactions. Note that in the case HLF, no TC-U-Abort indications are sent to the host.

- PPL Event Request (0x44)

  -The format of this message does not change. After an SS7 card has recovered from a host link failure, SCCP brings those subsystems in-service that were configured in-service before the host link failure occurred.

- System Configuration (0xAF)

  The Matrix Controller card already supports this message to enable/disable host link failure detection. The meaning of the `Response to Host link failure` field has been extended to allow configuring whether to take a SCCP/TCAP subsystem out-of-service when a host link failure occurs.

  This message now supports the SS7 card. When the Slot AIB is used for an SS7 card, an internal message is sent from the Matrix Controller card to the SS7 card to enable/disable the SS7 local host link failure detection feature, as well as, configure the host link failure detection timer for the SS7 card.

- System Configuration Query (0xB4)

  The matrix already supports this message. Changes to this message reflect changes to the *System Configuration* message.

Configuration Byte

Use configuration byte, 0x03, of the PPL component TCAP TUSI (0x70), to configure whether to initiate the TCAP restart procedure when a subsystem is out-of-service. See *TCAP TUSI (0x0070) (7-225)*.

SS7 Host Link Failure Detection Architecture

SS7 Host Link Failure Detection and Handling Logic

An SS7 local host is capable of detecting and managing a host link failure with an SS7 card and take subsystems out-of-service is needed. Previously in a CSP, if the CSP Matrix Series 3 Card detected any host link failure, it would automatically take all spans and channels out-of-service. Host link failure detection is now a separate process on an SS7 card. For more details, see *SS7 Host Link Failure Detection and Recovery (2-52)*.

Enhancements to Matrix Host Link Failure Handling

Because of recent enhancements, if the matrix controller local host detects a host link failure, there will be no impact on an SS7 host; no subsystems that reside on an SS7 card will be taken out-of-service. With software enhancements, only those subsystems that have been configured to reside on the matrix host will be taken out-of-service. Also these subsystems will be taken out-of-service only if the host has been configured to take the subsystem out-of-service when a host link failure occurs. In this case, the SCCP/TCAP subsystem out-of-service procedure applies.

SCCP/TCAP Subsystem Out-of-Service Procedure

The SCCP/TCAP module on an SS7 card can be instructed to take a subsystem out-of-service. As part of the SCCP subsystem out-of-service procedure, SCCP sends SSP (SubSystem Prohibited) to all the relevant point codes on the network.

If SCCP/TCAP module has been configured to initiate TCAP restart when a local subsystem is out-of-service (see *TCAP TUSI (0x0070) (7-225)*), a TCAP restart procedure will be initiated. In this case, all the active TCAP transactions for this local subsystem are aborted. TCAP Abort messages can be sent to the network when necessary according to the TCAP protocol. Note that the TC-U-Abort indication will not be sent to the host because the host link connection failed.

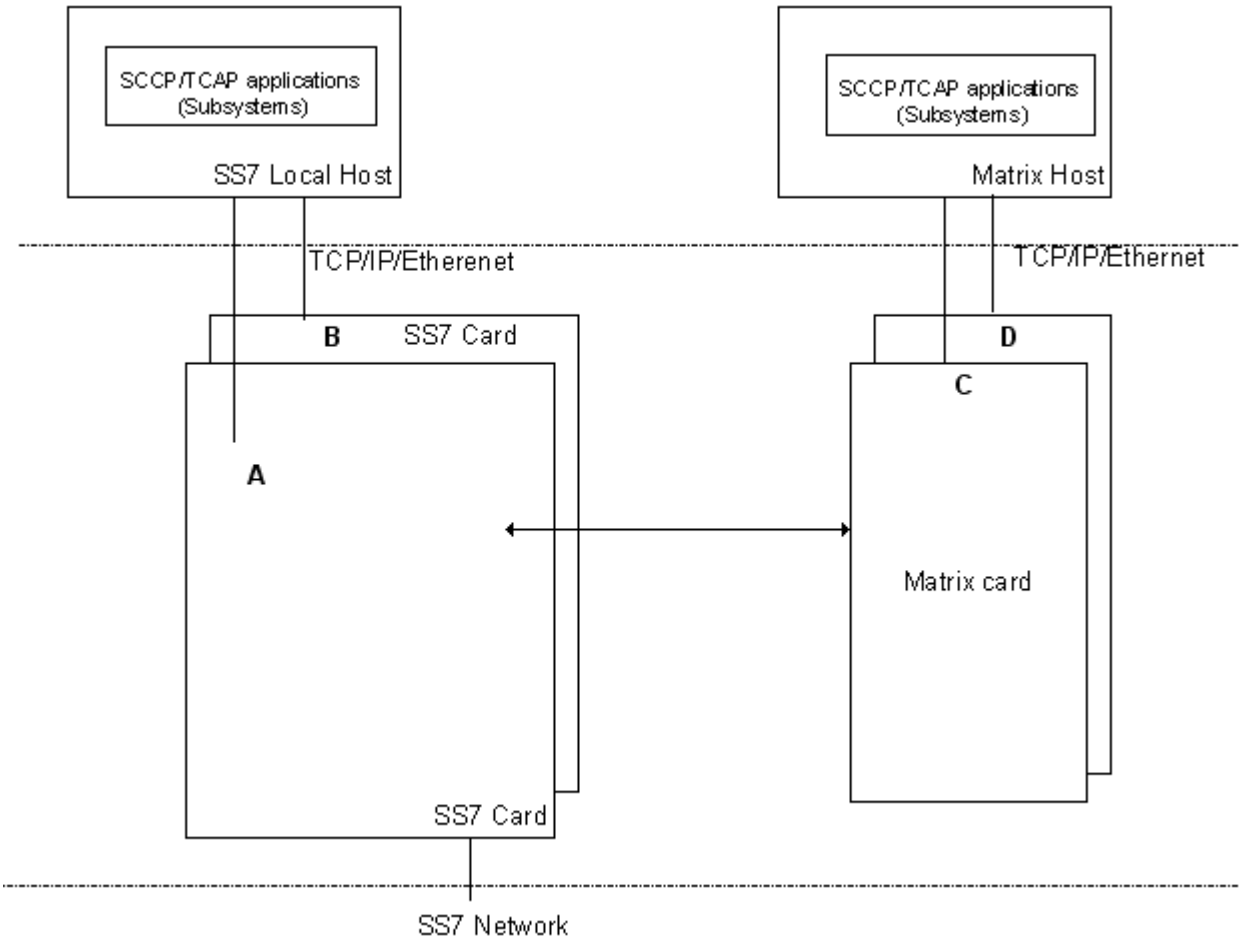
SCCP/TCAP Subsystem In-Service Procedure

As part of the SCCP subsystem in-service procedure, SCCP sends an SSA (SubSystem Allowed) message to the configured concerned point codes on the network.

Diagram of SS7 Host Link Failure Architecture

In the next diagram, A, B, C, and D are the host link failure detection points where the host link failure detection logic resides. Host link failure detection is set up as two separate processes: one for an SS7 card and one for a Matrix Controller card.

The TCAP and SCCP module is responsible for taking a subsystem out-of-service and initiates the TCAP restart procedure when a host link failure occurs.



SS7 Host Link Failure Detection and Recovery

Overview

The SS7 host link failure detection feature is disabled by default. To enable this feature you must send the *System Configuration* message to the matrix card with the SS7 card slot number. This message also configures the SS7 host link failure timer.

SS7 Host Link Failure Detection Logic

If the SS7 card does not receive any message from the host after four consecutive *Poll* messages, the CSP initiates a timer. If the CSP does not receive an message from the host before the timer expires, the CSP determines that a host link failure has occurred. The *System Configuration* message configures the Failure Confirmation Timer (Data[2,3]).

If no message has been received from the SS7 local host for the period of the timer (four *Poll* messages), a minor alarm (with card alarm type Host Link Error detected) is sent to the matrix host. See Alarm (0x00B9), *API Reference*. The SS7 host link failure timer then starts.

If the host link failure timer expires, the SS7 host link failure handling logic applies.

If the SS7 card receives any message from the local host before the host link failure timer expires, the host link is considered recovered. The host link failure timer is then stopped and an *Alarm Cleared* for the host link error is sent to the matrix host.

SS7 Host Link Failure Handling Logic

When the SS7 card detects a local host link failure, a major card alarm (with alarm type Ethernet link failure) is sent to the matrix host. See Alarm (0x00B9), *API Reference*. If the SS7 card is in a redundant mode, the card sends an internal message to inform its mate about the host link failure status. This status change will be reflected in the next *Poll* message the SS7 mate sends to its host. Depending on the SS7 card status, one of the following applies:

- If the card is single, and the host has been configured to take the subsystem out-of-service when a host link failure is detected, then all the subsystems that reside on the SS7 local host are taken out-of-service. The SCCP/TCAP subsystem out-of-service procedure applies. See *SCCP/TCAP Subsystem Out-of-Service Procedure (2-50)*.

- Assuming redundant SS7 cards are in the active/standby mode, if the active SS7 card loses the TCP connection to its local host, the SS7 card checks whether its mate standby SS7 card is in host link failure condition. If the standby card is still connected to its local host, the card expects the local host to initiate a switchover by sending a *Become Active* request to the standby card. No further action is taken by the SS7 card. If the host does not initiate the switchover within 10 seconds or the standby card loses its host link connection within 10 seconds, the SS7 takes the local subsystems out-of-service. The SCCP/ TCAP subsystem out-of-service procedure applies. See *SCCP/TCAP Subsystem Out-of-Service Procedure (2-49)*.
- Assuming redundant SS7 cards are in the active/standby mode, if both active and standby SS7 cards lose the TCP connections to their local host(s), all the subsystems that have been configured to reside on the SS7 local host is taken out-of-service. This assumes also that the SS7 card has been configured to take subsystems out-of-service in the event of a host link failure. The SCCP/TCAP subsystem out-of-service procedure applies.
- Assuming the redundant SS7 cards are in the synchronizing mode, if the active SS7 card loses the TCP connection to its local host then all the subsystems residing on the SS7 local host are taken out-of-service. This assumes also that the SS7 card has been configured to take subsystems out-of-service in the event of a host link failure. The SCCP/TCAP Subsystem out-of-service procedure applies.
- If the redundant SS7 cards are in the active/standby mode or synchronizing mode, and the standby SS7 card loses the TCP connection to its local host, no further action is taken.

Matrix Host Link Failure Recovery

After the matrix host link is recovered, the subsystem on the host is presumed to be in its previously-configured state. If the subsystem was in-service before the host link failure, SCCP goes through local subsystem in-service procedure as if it receives N_STATE request from host.

SS7 Host Link Failure Recovery

If the SS7 card receives any message from its local host, then the SS7 host link is considered recovered. If the SS7 card was in host link failure condition, *Alarm Cleared* with alarm types, Ethernet Host Link Error Detected (0x10) and Ethernet Link Failure (0x22) is sent to the matrix host. See *Alarm Cleared 0x00C1, API Reference.*

If the SS7 card is in the redundant mode, it informs its mate SS7 card about the host link recovery. This is reflected in the next *Poll* message from both SS7 cards if polling is enabled.

If the subsystem was brought in-service by the host before the host link failure, SCCP goes through the local subsystem in-service procedure as if it receives N_STATE request from host.

SS7 Polling and Host Link Failure

Overview

Polling has to be enabled to enable SS7 host link failure detection. The polling scheme allows the host to control polling from the SS7 card. That is, the SS7 card sends *Poll* messages to its local host by default, and the host can stop and start polling as well as change the polling interval.

After receiving a *Poll Interval* message from its local host, the SS7 card stops the polling or changes the poll interval accordingly. The SS7 card starts to send a *Poll* message to the local host after processing a *Poll Request* from the SS7 local host. When using SS7 redundancy, these messages are sent to the active SS7 card. The SS7 card propagates the message to its standby card. If these messages are sent to the standby card, a NACK (0x14) is sent back.

Active/Standby Mode

When SS7 is running in the active/standby mode, if a host link failure occurs on one SS7 card, the mate SS7 card is still connected to the local host. The mate SS7 card sends a *Poll* message to the host indicating the host link failure. While an SS7 card pair is in the active/standby mode, if the active SS7 card loses its connection to the local host and the standby SS7 card is still connected to the local host, the host software is assumed to send a *Become Active* message to the standby SS7 card to initiate an SS7 switchover. The host is aware of the active SS7 host link status from the *Poll* message from the standby card. If the host does not initiate the switchover within 10 seconds, the SS7 card will take the local subsystems out of service. If the host does not initiate the switchover within 10 seconds because the standby card has also lost its host link connection within that 10 seconds, the SS7 card will take the local subsystems out of service. The SCCP/TCAP subsystem out of service procedure applies.

SS7 Signaling Stack Congestion Control

Overview

The Signaling Stack Congestion Control feature prevents the SS7 card from failing due to call traffic overload. Congestion Control is enabled by default and Dialogic recommends that you do not disable it.

Automatic congestion control monitors inbound and outbound traffic (from the CSP perspective). Congestion (at the CSP) is not invoked as a result of an adjacent CSP entering a congested state. When an adjacent platform enters a congested state, Dialogic-initiated calls to that platform will fail. A Release Message (REL) will be received in the backward direction in response to the Initial Address Message (IAM). The cause parameter within the REL message will indicate congestion. It is the host's responsibility to examine the cause parameter in the *Channel Released with Data API*, and throttle (stop) traffic for a host-determined amount of time.

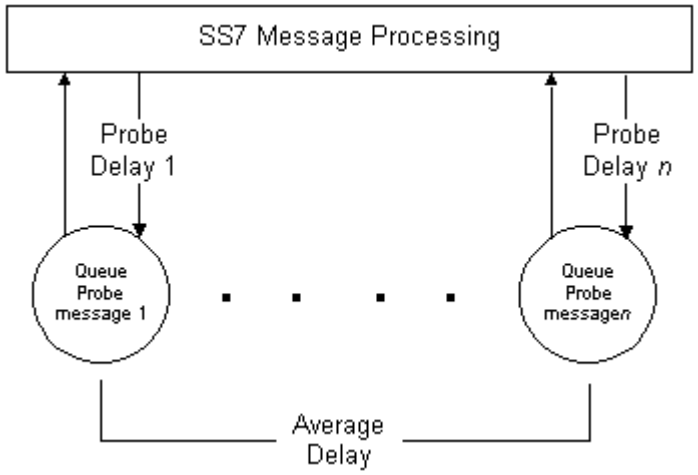
If the CSP encounters congestion in the outbound direction, attempts to *Outseize* will result in a negative acknowledgment of 0x0C (Outbound Congestion). In this case no IAM is sent to the network.

Congestion Levels

The level of congestion on each signaling stack is monitored by the Signaling Procedure Control (SPRC) component. Internal “Queue Probe” messages are periodically sent to measure the amount of delay (backup) in the SS7 message queue.

Each time a “Queue Probe” is returned to SPRC, the delay for that probe is recorded, as well as an average delay for ***n*** number of probes. If either delay reaches the configured threshold, a congestion condition is determined to exist and action is taken to relieve the congestion.

Figure 2-12 Calculating Congestion Level



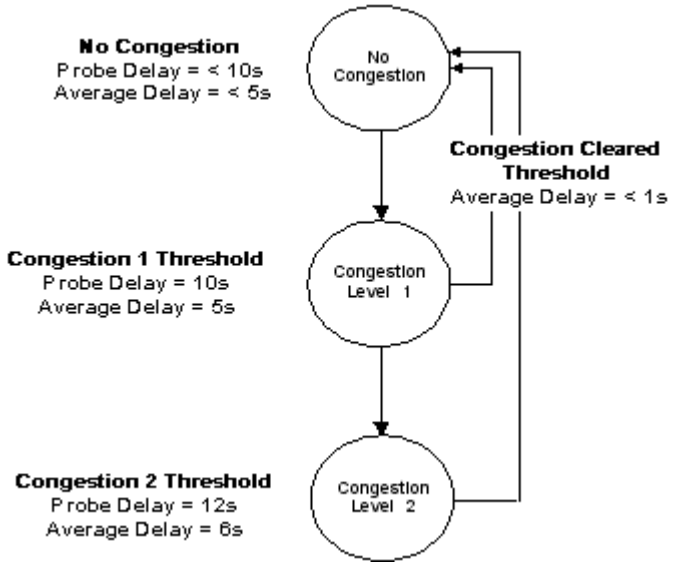
Congestion Status

There are three congestion statuses:

- No Congestion
- Level 1
- Level 2

The transition between congestion statuses and the associated thresholds with default settings is shown in the next figure.

Figure 2-13 Congestion Levels



Default Values

The default threshold values are as follows:

Congestion Level 1

- Probe Delay: 10 seconds
- Average Delay: 5 seconds

Congestion Level 2

- Probe Delay: 12 seconds
- Average Delay: 6 seconds

Congestion Cleared

- Average Delay: < 1 seconds

The default values for congestion Level 2 are not configurable. The number of probes used to calculate the Average Delay is 10 and the time between probes is one second.

System Busy, Host Overload, and Exchange Overload

A Congestion Indication may also be sent by MTP for both ANSI and ITU-TS. In this case, a General alarm of SS7 Signaling Network Congestion (0x0E) is sent to the host. It is the responsibility of the host to implement the proper reaction for Congestion Indications.

For ANSI, the congestion level byte in the alarm has a value from 0–3, where 0 represents no congestion.

For ITU, the congestion level byte is always 1 [no alarm received after a given amount of time (for example, ITU-TS T30) indicates no congestion]. For ITU, see Q.767 D.2.11. For ANSI, see T1.113-1992 2.11.

The CSP may generate a System Busy alarm when it is processing a heavy call load. In addition, there may times when the host application experiences overload conditions. In either of these cases, the host should initiate procedures to try to reduce incoming traffic to the CSP. ISUP supports a mechanism for presenting a level of exchange overload to other connected SS7 exchanges as described below.

After detecting a System Busy *Alarm* or host application overload, the host should insert the Automatic Congestion Level parameter into all outgoing REL messages (for a description, see ANSI T1.113.4-1992 section 2.11, Annex D and ITU Q.767 Section D.2.12.) This parameter relays a level of exchange overload to the other exchanges. The other exchanges in turn, attempt to reduce the volume of traffic to the CSP.

Upon receiving an *Alarm Cleared* message or if the host application becomes less overloaded, the host can discontinue including the Automatic Congestion Level parameter in the REL messages. The distant exchanges then return to normal traffic to the CSP after a predetermined period of time.

Just as the host should be able to generate REL messages with the Automatic Congestion Level parameter, the host should also scan incoming *Channel Released With Data* messages for Automatic Congestion Level parameters. If Automatic Congestion Level parameters are received in the incoming REL messages, the host should attempt to reduce the volume of traffic to the DPC that generated the Automatic Congestion Level parameter until the DPC stops inserting Automatic Congestion Level parameters into its outgoing REL messages (see ANSI T1.113.4 Annex D, or ITU Q.767 Section D2.12).

CSP Response to Congestion

Overview

This section describes the CSP response when the congestion level reaches Level 1 or Level 2 on an SS7 card.

Congestion Level 1

If congestion Level 1 is reached, an *Alarm* message is sent to the host and action is taken on the congested stack.

The *Alarm* message in response to congestion Level 1 includes the following information:

Severity: Major (0x01)

Entity: General (0x01)

Alarm Number: SS7 Signaling Stack Congestion (0x16)

Data[0] Stack ID

Data[1] Direction

0x00	Incoming
0x01	Outgoing

Data[2] Status

0x01	Congestion Level 1
0x02	Congestion Level 2

The action the CSP takes in response to congestion depends on whether the call is incoming or outgoing and whether the stack is configured for ITU or ANSI and ISUP or TUP.

ITU/ANSI Incoming Calls

If congestion Level 1 is reached, the call is not processed and a *Release* message is sent to the call originator with a cause of Congestion. An optional parameter of *Automatic Congestion Control* is included, with the congestion level as defined in the protocol.

TUP Incoming Calls

If congestion Level 1 is reached, a Backward SETUP Indication message is sent to the call originator. The Return Cause reason (Congestion Type) is configurable with Config Byte 20 of the TUP CPC component.

When the originator receives this message it sends a Clear Forward message. After receiving Clear Forward message, the CSP experiencing congestion will send an Automatic Congestion Control message followed by a Release Guard message.

Outgoing Calls

If congestion Level 1 is reached, L3P or L3P TUP sends an Access Denied indication to the host with a Cause Code of 0x0C.

Congestion Level 2

If congestion Level 2 is reached, all new call MSUs (IAM, IAI, SAM, SAO) are discarded by MTP3. The host is sent an *Alarm* message indicating the change in Congestion Status to Level 2 (0x02). All new call MSUs are discarded until the congestion cleared threshold is reached and the stack returns to a No Congestion status.

From congestion Level 2, you can only go to a No Congestion status. The Congestion Status cannot go from Level 2 to Level 1. When the Congestion Cleared threshold is reached and maintained for ***n*** number of probes, the host is sent an *Alarm Cleared* message and normal call processing resumes. See *Congestion Cleared (2-61)* for more information.

> **Important!** Congestion Level 2 can only be reached for incoming calls. Outgoing call requests are rejected if congestion Level 1 is reached.

Congestion Cleared

When the congestion condition is reduced below the Congestion Cleared threshold and remains below that level for a specified number of probes, an *Alarm Cleared* message is sent to the host and normal call processing resumes on the stack.

The *Alarm Cleared* message includes the following information:

Severity: Major (0x01)

Entity: General (0x01)

Alarm Number: SS7 Signaling Stack Congestion (0x16)

Data[0] Stack ID

Data[1] Direction

0x00	Incoming
0x01	Outgoing

Data[2] Congestion Status

0x00	Congestion Cleared

Modifying Congestion Control

This section describes modifications you can make to the default Congestion Control implementation. Congestion levels are monitored by the ISUP SPRC (0x13) or TUP SPRC (0x53) components. To change the default implementation of congestion control, modify the following PPL Config Bytes or Timers as required:

- Congestion Level 1 Probe Delay Threshold - Incoming Calls

  PPL Config Byte 0x29
- Congestion Level 1 Probe Delay Threshold - Outgoing Calls

  PPL Config Byte 0x2D

  See for default values
- Congestion Level 1 Average Delay Threshold - Incoming Calls

  PPL Config Byte 0x2A
- Congestion Level 1 Average Delay Threshold - Outgoing Calls

  PPL Config Byte 0x2E
- Amount of time between probes

  PPL Timer 2
- The number of probes used to calculate the Average Delay

  PPL Config Byte 0x28
- Congestion Cleared Threshold - Incoming Calls

  PPL Config Byte 0x2B

  Number of probes to determine Incoming Congestion Cleared

  PPL Config Byte 0x2C

- Congestion Cleared Threshold - Outgoing Calls

  PPL Config Byte 0x2F

  Number of probes to determine Outgoing Congestion Cleared

  PPL Config Byte 0x30

Configuring SS7 Virtual Spans

Purpose

The Virtual Spans software allows configuration of virtual T1, E1, and J1 cards in the CSP. This feature meets all standard software requirements for Dialogic and does not affect connectivity or performance.

The Virtual Spans feature is common to the Dialogic SS7, ISDN, DASS2, and DPNSS protocols. By configuring virtual spans on the CSP, physical links or voice circuits with their voice and data traffic can be terminated on another CSP outside the CSP platform. The CSP simply terminates control channels for the SS7 signaling links, ISDN channels, and provides management of SS7 CICS and ISDN B Channels. No difference is apparent to the CSP between virtual and physical spans installed in the chassis.

Before you begin

Virtual cards function in the same way as physical cards in all respects. Insertion and removal of cards will occur logically for the CSP without interruption to service, using the *Card Status Query* and *Card Status Report* messages.

The following card types represent the four, eight, and 16-span versions of the E1, T1, and J1 virtual cards:

0xD0 Virtual T1 4 Span
0xD1 Virtual T1 8 Span
0xD2 Virtual T1 16 Span
0xD3 Virtual E1/J1 4 Span
0xD4 Virtual E1/J1 8 Span
0xD5 Virtual E1/J1 16 Span

Typical Operation of Virtual Spans

The following functions apply to typical operation of the Virtual Spans feature:

- A CSP 2090 supports up to 2048 ports and can function with any combination of virtual and physical spans.
- Configuration of virtual spans is done in the same manner as configuration of physical spans.
- Call processing also functions as it would in a CSP system that had only physical spans installed.
- Virtual spans can be configured on local SS7 server nodes or on remote nodes.

- Since virtual slots in the CSP are each assigned a Card Type as if they were installed as physical cards, both matrix controller diagnostics and alarms function normally.
- Messaging occurs normally for SS7, ISDN, DASS2 and DPNSS protocols.
- Product Licensing is supported.
- The same messages that reset a physical card's tag configuration also reset the equivalent virtual card. You must send the *Tag Configuration* (0x00D0) message to retag the virtual card. Refer to the *Tag Configuration (0x00D0)* message in the *API Reference* for a list of configuration messages that reset the tag configuration of physical cards.

Configuration of Virtual Spans

The Virtual Span configuration sequence incorporates the API messages as shown in the table below:

Step	API Message	Action
1	Assign Logical Span ID	De-assign all of the spans and associated channels in the system. This will remove all virtual slots. These virtual slots must then be configured again.
2	Virtual Slot Configure	Add the virtual slots to the system.
3	Assign Logical Span ID	Assign logical spans to the appropriate slots and offsets.
4	Messages from Standard Configuration Sequence (SS7, ISDN...)	See *API Developer's Guide: Line Cards* and *Developer's Guide: Overview.*
5	Service State Configure	Bring all spans and associated channels into service.
6	Virtual Span Control	Inform the CSP that the Virtual Spans are operational. (default =non-operational)

Virtual Card Acts as Physical Card

The *Virtual Card Configure* message allows the CSP to function as if a "real" physical card has been installed. The host receives the *Card Status Report* as if the virtual card actually existed. It monitors each card type as real (physical) so that the virtual card, whether it is T1, E1, or J1, can only be inserted or removed through transmission of the

Virtual Card Configure message.The *Virtual Span Control* message supervises the actual spans that the virtual card controls. See the *API Reference* for a detailed description of these messages.

Redundancy

Matrix Redundancy

The virtual card addition and the virtual span status is mirrored on the standby CSP Matrix Series 3 Card. Therefore, in the event of the active matrix controller failure the standby will take over, maintaining the virtual card and span status. If there are active calls up at the time of the failure, they will still be maintained on the newly active card.

N+1 Redundancy

N+1 redundancy is not a valid redundancy option for a virtual card.

SS7 and ISDN Redundancy

SS7 and ISDN redundancy mechanisms work exactly the same as they did before, using virtual spans. The only possible implication is that the protocol stacks will query the virtual span state (internal to the system) and will get the latest virtual span status that was configured by the Host using the *Virtual Span Control* message.

Virtual Spans Example Configuration File

The following is an example configuration file that utilizes the *Virtual Card Configure* and the *Virtual Span Control* messages. It sets up two SS7 ANSI stacks and controls CICs on the virtual spans.

The high level CSP configuration is:

Slot 0 SS7 card

Slot 1 Virtual 8-span card

Slot 3 T-ONE 8-span card

Spans 0-3 are assigned and carry the signaling links. Spans 8 to 15 are the spans where the CICs are assigned.

```
'De-Assign all Logical span ids
00 0d 00 a8 00 00 ff 00 01 11 04 ff ff ff ff

'Virtual card configure (add 8 span T1)
'D0 4    Span T1
'D1 8    Span T1
'D2 16  Span T1
'D3 4   Span E1
```

```
'D4 8   Span E1
'D5 16  Span E1
00 0d 00 e0 00 00 FF 00 00  01 01 01  02 01  D1

'Assign logical spans for the links
 00 0d 00 a8 00 00 ff 00 01 11 04 00 00 03 00
 00 0d 00 a8 00 00 ff 00 01 11 04 00 01 03 01
 00 0d 00 a8 00 00 ff 00 01 11 04 00 02 03 02
 00 0d 00 a8 00 00 ff 00 01 11 04 00 03 03 03
'Assign spans 8-15 to the virtual slots
 00 0d 00 a8 00 00 ff 00 01 11 04 00 08 01 00
 00 0d 00 a8 00 00 ff 00 01 11 04 00 09 01 01
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0a 01 02
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0b 01 03
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0c 01 04
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0d 01 05
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0e 01 06
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0f 01 07

'Set span format (ESF, B8ZS, clear channel, 133ft)
00 0d 00 a9 00 00 ff 00 01 0c 02 00 00 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 01 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 02 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 03 52 06

'Set span filter Characteristics (1 second CGA Cleared)
00 0e 00 cd 00 00 ff 00 01 0c 02 00 00 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 01 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 02 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 03 01 00 64


'****************************

'SS7 CONFIGURATION STARTS HERE

'****************************


'-------------------------------------------------------
   -----------------
'Configure First SS7 Stack (A)
'-------------------------------------------------------
   -----------------

'Set Signaling Stack Configure A
'Set our OPC to 00000123 and set the 3 modules to ANSI
00 16 00 5c 00 00 ff 00 01 21 02 00 00 00 00 01 23 03 01
   00 02 00 03 00

'Define a Link Set going to 00000987 ID 0
```

```
00 0f 00 5d 00 00 ff 00 01 1e 02 00 00 00 00 09 87

'Set the signaling link configuration (SLC 0, Link Id 0,
   Link set 0)
00 14 00 5e 00 00 ff 00 02 1f 03 00 00 00 0d 03 00 00 00
   00 00 00

'Set the Signaling Route Configure for A
'Define a Route to 00 00 09 87 using only Link Set 0
00 14 00 5f 00 00 ff 00 01 20 05 00 00 00 00 00 00 00 09
   87 00 00

' Assign DPC-CIC 00000987=0000-060 for A
00 14 00 6a 00 00 ff 00 01 14 07  00    00 00   00 08 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 20   00 09 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 40   00 0a 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 60   00 0b 00
   18  00 00 09 87


'-----------------------------------------------------------------------

' Configure Second SS7 Stack (B)

'-----------------------------------------------------------------------

' Set Signaling Stack Configure B
' Set our OPC to 00000987 and set the 3 modules to ANSI
00 16 00 5c 00 00 ff 00 01 21 02  00 01 00 00 09 87 03 01
   00 02 00 03 00

' Define a Link Set going to 00000123 ID 1
00 0f 00 5d 00 00 ff 00 01 1e 02 01 01 00 00 01 23

' Set the Signaling Link Configuration (SLC 0, Link Id 1,
   Link set 0)
00 14 00 5e 00 00 ff 00 02 1f 03 01 01 01 0d 03 00 01 00
   00  00 00

' Set the Signaling Route Configure for B
' Define a Route to 00 00 01 23 using only Link Set 1
00 14 00 5f 00 00 ff 00 01 20 05 01 00 01 00 01 00 00 01
   23 01 00

' Assign DPC-CIC 00000123=0000-0060 for B
00 14 00 6a 00 00 ff 00 01 14 07 01 00 00 00 0c 00 18 00
   00 01 23
00 14 00 6a 00 00 ff 00 01 14 07 01 00 20 00 0d 00 18 00
   00 01 23
```

```
00 14 00 6a 00 00 ff 00 01 14 07 01 00 40 00 0e 00 18 00
   00 01 23
00 14 00 6a 00 00 ff 00 01 14 07 01 00 60 00 0f 00 18 00
   00 01 23


'*****************************

' SS7 CONFIGURATION ENDS HERE

'*****************************


'Bring spans into service
00 0c 00 0a 00 00 ff 00 01 0c 02 00 00 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 01 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 02 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 03 f0

'Bring virtual span into service (optional)
00 0c 00 0a 00 00 ff 00 01 0c 02 00 08 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 09 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0a f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0b f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0c f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0d f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0e f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0f f0

'Bring the Links in Service
00 0d 00 0a 00 00 ff 00 01 09 02 00 00 f0 00
00 0d 00 0a 00 00 ff 00 01 09 02 01 01 f0 00

'Bring channels into service
00 12 00 0a 00 00 ff 01 02 0d 03 00 08 00 0d 03 00 08 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 09 00 0d 03 00 09 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0a 00 0d 03 00 0a 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0b 00 0d 03 00 0b 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0c 00 0d 03 00 0c 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0d 00 0d 03 00 0d 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0e 00 0d 03 00 0e 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0f 00 0d 03 00 0f 17
   f0
```

```
'Virtual Span Control
'Tell the switch the spans are alive
00 0e 00 e2 00 00 FF 00 01 0C 02 00 08 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 09 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0a 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0b 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0c 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0d 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0e 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0f 00 01 00
```

Host-Controlled Continuity

The Continuity Check is performed for the voice circuits (CICs). This involves testing whether a tone has been properly transmitted at one end and been received at the other on the CIC for which the call is intended. If the Continuity Check passes, then the call continues. If it does not, the call is released.

Since virtual T1, E1, and J1 cards function the same way real cards would in the CSP, the CSP will automatically attempt a Continuity Check for a CIC when an IAM is received. However, since no actual span exists, this Continuity Check will fail. To prevent this, the automatic Continuity Check is disabled in the CSP. In its place Dialogic uses the “Host-controlled Continuity”. Using this feature, the host can initiate a Continuity Check manually by choice only.

Call Flows for Virtual Spans

Refer to *Call Flows for Virtual Spans (2-80)*

Configuring SS7 10K Virtual Channels

Overview

The 10K Virtual Channels feature (sometimes referred to as Virtual CICs) allows configuration of virtual T-ONE and E-ONE/J-ONE cards in the CSP. This enhancement meets all standard software requirements for Dialogic and does not affect connectivity or performance.

The 10K Virtual Channels feature is supported by the SS7 Series 3 protocol. By configuring virtual channels on the CSP, physical links or voice circuits with their voice and data traffic can be terminated on another switch outside the CSP platform. The CSP simply terminates control channels for the SS7 signaling links and provides management of SS7 Circuit Identification Codes (CICs). No difference is apparent to the CSP between virtual and physical channels installed in the chassis.

Before You Begin

Important! Please note the following:

- A single CSP supports the 10K Virtual Channels feature for this release.
- A maximum of 12,000 T1 or E1/J1 channels can be supported per CSP. There are the 2,000 physical channels and up to 10,000 virtual channels available for licensing (See *Licensing of Virtual Channels* below.)
- The SS7 Series 3 card supports the 10K Virtual Channels feature.
- The SS7 PQ card does not support the 10K Virtual Channels feature.
- You cannot connect virtual channels over the Exnet® Ring. In order for a call to be connected over the Exnet® Ring, the channels must be terminated locally (that is, on the CSP or Exnet Connect® card).

Licensing of Virtual Channels

The licensing of Virtual Channels, based on the CSP chassis serial number, can be from an initial 2,000 up to a maximum of 10,000. To use the 10K Virtual Channels feature, you must configure Virtual Channels in the CSP. The CSP does not distinguish between Virtual Channels and physical channels.

The licensing of Virtual Channels, based on the CSP chassis serial number, can be from an initial 2,000 up to a maximum of 10,000 at 2,000 increments. To enable more than 2000 Virtual Channels you

require at least two licenses, one to enable the initial virtual 2000 channels, and a second license to incrementally increase to greater than 2000 Virtual Channels.

The following licensing options are available by model number:

- Initial license for 2,000 Virtual Channels
- Incremental 2,000 Virtual Channels
- Incremental 4,000 Virtual Channels
- Incremental 6,000 Virtual Channels
- Incremental 8,000 Virtual Channels

Activating Licensing of Virtual Channels

To activate any new feature, including 10K Virtual Channels, you must use a Product License Key, which Dialogic provides when you purchase the Product License. The key is unique and encrypted.

For details, refer to *Downloading License Keys to the CSP* in the *Licensing Overview* chapter in the *Developer's Guide: Overview* and the *Product License Download* message (0x0079) in the *API Reference.*

Important! If you have two licenses for an upgrade from 2,000 channels to 10,000 channels, use the 10,000 channel license. If you download the 2,000 license after a 10,000 license was already downloaded, the CSP will revert to 2,000 channels.

T-ONE and E-ONE/J-ONE Virtual Cards

Virtual cards function in the same way as physical cards in all respects. Insertion and removal of cards will occur logically for the switch without interruption to service, using the *Card Status Query* and *Card Status Report* messages.

Two card types represent the T-ONE and E-ONE/J-ONE virtual cards. These are:

- 0xD2 Virtual T-ONE, 16 span
- 0xD5 Virtual E-ONE/J-ONE, 16 Span

Important! You cannot use *Service State Configure* to take virtual cards out-of-service.

Expanded Virtual Card Slots

To support the expanded number of channels, it is necessary to extend the range of slot values for the *Virtual Slot Configuration* message.

A total of 32 virtual slot values 64-95 (0x40-0x5F) have been added, to allow for the 10k channels to be mapped without the current requirement of mapping a virtual slot to an empty slot in the CSP chassis.

The following tables break down the number virtual cards supported for the additional virtual channels and spans.

Virtual E-ONE/J-ONE Cards: Spans and Channels

Channels Licensed	E-ONE or J-ONE Cards	Virtual Channels	Virtual Spans
2,000	4	2,048	64
4,000	8	4,096	128
6,000	12	6,144	192
8,000	16	8,192	256
10,000	*20	10,080	315

Important! *The 20th virtual card can only have 11 spans assigned to it to get the maximum of 315 virtual E1/J1 spans allowed.

Virtual T-ONE Cards: Spans and Channels

Channels Licensed	T-ONE Cards	Virtual Channels	Virtual Spans
2,000	6	2,304	96
4,000	11	4,224	176
6,000	16	6,144	256
8,000	22	8,448	352
10,000	*27	10,080	420

Important! *The 27th virtual card can only have 4 spans assigned to it to get the maximum of 420 virtual T1 spans allowed.

A fully configured system provides the following spans, channels and virtual cards at capacity.

Virtual T-ONE and E-ONE/J-ONE Card Span and Channel Capacities

Virtual Card Type	Number of Spans per Virtual Card	Number of Channels per Virtual Span	Number of Channels per Virtual Card	Total Number of Virtual Spans	Minimum Number of Virtual Cards at Capacity
T-ONE (0xD2)	16	24	384	420	27
E-ONE/J-ONE (0xD5)	16	32	512	315	20

Typical Operation of Virtual Spans

The following functions apply to typical operation of the Virtual Spans feature:

- A CSP supports up to 2048 ports and can function with any combination of virtual and physical spans.
- Configuration of virtual spans is done in the same manner as configuration of physical spans.
- Call processing also functions as it would in a CSP system that had only physical spans installed.
- Virtual spans can be configured on local SS7 Server Nodes or on remote nodes.
- Since virtual slots in the switch are each assigned a Card Type as if they were installed as physical cards, both matrix diagnostics and alarms function normally.
- Messaging occurs normally for SS7.

Configuration of Virtual Spans

The Virtual Span configuration sequence incorporates the API messages as shown in the table below:

Step	API Message	Action
1	*Assign Logical Span ID*	De-assign all of the spans and associated channels in the system. This will remove all virtual slots. These virtual slots must then be configured again.
2	*Virtual Card Configure*	Add the virtual slots to the system.
3	*Assign Logical Span ID*	Assign logical spans to the appropriate slots and offsets.
4	Messages from Standard Configuration Sequence (SS7)	See *API Developer's Guide: Overview*

Step	API Message	Action
5	*Service State Configure*	Bring all spans and associated channels into service.
6	*Virtual Span Control*	Inform the switch that the Virtual Spans are operational. (default =non-operational)

Virtual Card Acts as Physical Card

The *Virtual Card Configure* message allows the switch to function as if a “real” physical card has been installed. The host receives the *Card Status Report* as if the virtual card actually existed. It monitors each card type as real (physical) so that the virtual card, whether it is T-ONE or E-ONE/J-ONE, can only be inserted or removed through transmission of the *Virtual Card Configure* message.

The *Virtual Span Control* message supervises the actual spans that the virtual card controls. See the *API Reference* for a detailed description of these messages.

Redundancy

Matrix Redundancy

The virtual card addition and the virtual span status is mirrored on the standby matrix. Therefore, in the event of the active matrix failure the standby will take over, maintaining the virtual card and span status. If there are active calls up at the time of the failure, they will still be maintained on the newly active card.

N+1 Redundancy

N+1 redundancy is not a valid redundancy option for a virtual card.

SS7 Redundancy

SS7 redundancy mechanisms work exactly the same as they did before, using virtual spans. The only possible implication is that the protocol stacks will query the virtual span state (internal to the system) and will get the latest virtual span status that was configured by the Host via the *Virtual Span Control* message.

Virtual Spans Example Configuration File

The following is an example configuration file that utilizes the *Virtual Card Configure* and the *Virtual Span Control* messages. It sets up two SS7 ANSI stacks and controls CICs on the virtual spans.

The high level CSP configuration is:

Slot 0 SS7 Series 3 card

Slot 1 Virtual 16-span card

Slot 3 T-ONE 16-span card

Spans 0-3 are assigned and carry the signaling links. Spans 0 to 15 are the spans where the CICs are assigned.

```
'De-Assign all Logical span ids
00 0d 00 a8 00 00 ff 00 01 11 04 ff ff ff ff

'Virtual card configure (add 16 span T1)
'D2 16  Span T1
'D5 16  Span E1/J1
00 0d 00 e0 00 00 FF 00 00  01 01 01  02 01  D2

'Assign logical spans for the links
 00 0d 00 a8 00 00 ff 00 01 11 04 00 00 03 00
 00 0d 00 a8 00 00 ff 00 01 11 04 00 01 03 01
 00 0d 00 a8 00 00 ff 00 01 11 04 00 02 03 02
 00 0d 00 a8 00 00 ff 00 01 11 04 00 03 03 03
'Assign spans 8-15 to the virtual slots
 00 0d 00 a8 00 00 ff 00 01 11 04 00 08 01 00
 00 0d 00 a8 00 00 ff 00 01 11 04 00 09 01 01
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0a 01 02
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0b 01 03
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0c 01 04
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0d 01 05
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0e 01 06
 00 0d 00 a8 00 00 ff 00 01 11 04 00 0f 01 07

'Set span format (ESF, B8ZS, clear channel, 133ft)
00 0d 00 a9 00 00 ff 00 01 0c 02 00 00 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 01 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 02 52 06
00 0d 00 a9 00 00 ff 00 01 0c 02 00 03 52 06

'Set span filter Characteristics (1 second CGA Cleared)
00 0e 00 cd 00 00 ff 00 01 0c 02 00 00 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 01 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 02 01 00 64
00 0e 00 cd 00 00 ff 00 01 0c 02 00 03 01 00 64

'****************************
'SS7 CONFIGURATION STARTS HERE
'****************************
```

```
'-------------------------------------------------------------
   -----------------
'Configure First SS7 Stack (A)
'-------------------------------------------------------------
   -----------------

'Set Signaling Stack Configure A
'Set our OPC to 00000123 and set the 3 modules to ANSI
00 16 00 5c 00 00 ff 00 01 21 02 00 00 00 00 01 23 03 01
   00 02 00 03 00

'Define a Link Set going to 00000987 ID 0
00 0f 00 5d 00 00 ff 00 01 1e 02 00 00 00 00 09 87

'Set the signaling link configuration (SLC 0, Link Id 0,
   Link Set 0)
00 14 00 5e 00 00 ff 00 02 1f 03 00 00 00 0d 03 00 00 00
   00 00 00

'Set the Signaling Route Configure for A
'Define a Route to 00 00 09 87 using only Link Set 0
00 14 00 5f 00 00 ff 00 01 20 05 00 00 00 00 00 00 00 09
   87 00 00

' Assign DPC-CIC 00000987=0000-060 for A
00 14 00 6a 00 00 ff 00 01 14 07  00    00 00   00 08 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 20   00 09 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 40   00 0a 00
   18  00 00 09 87
00 14 00 6a 00 00 ff 00 01 14 07  00    00 60   00 0b 00
   18  00 00 09 87

'-------------------------------------------------------------
   -----------------
' Configure Second SS7 Stack (B)
'-------------------------------------------------------------
   -----------------
' Set Signaling Stack Configure B
' Set our OPC to 00000987 and set the 3 modules to ANSI
00 16 00 5c 00 00 ff 00 01 21 02  00 01 00 00 09 87 03 01
   00 02 00 03 00

' Define a Link Set going to 00000123 ID 1
00 0f 00 5d 00 00 ff 00 01 1e 02 01 01 00 00 01 23

' Set the Signaling Link Configuration (SLC 0, Link Id 1,
   Link Set 0)
00 14 00 5e 00 00 ff 00 02 1f 03 01 01 01 0d 03 00 01 00
   00  00 00
```

```
' Set the Signaling Route Configure for B
' Define a Route to 00 00 01 23 using only Link Set 1
00 14 00 5f 00 00 ff 00 01 20 05 01 00 01 00 01 00 00 01
   23 01 00

' Assign DPC-CIC 00000123=0000-0060 for B
00 14 00 6a 00 00 ff 00 01 14 07 01 00 00 00 0c 00 18 00
   00 01 23
00 14 00 6a 00 00 ff 00 01 14 07 01 00 20 00 0d 00 18 00
   00 01 23
00 14 00 6a 00 00 ff 00 01 14 07 01 00 40 00 0e 00 18 00
   00 01 23
00 14 00 6a 00 00 ff 00 01 14 07 01 00 60 00 0f 00 18 00
   00 01 23

'****************************
' SS7 CONFIGURATION ENDS HERE
'****************************

'Bring spans into service
00 0c 00 0a 00 00 ff 00 01 0c 02 00 00 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 01 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 02 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 03 f0

'Bring virtual span into service (optional)
00 0c 00 0a 00 00 ff 00 01 0c 02 00 08 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 09 f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0a f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0b f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0c f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0d f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0e f0
00 0c 00 0a 00 00 ff 00 01 0c 02 00 0f f0

'Bring the Links in Service
00 0d 00 0a 00 00 ff 00 01 09 02 00 00 f0 00
00 0d 00 0a 00 00 ff 00 01 09 02 01 01 f0 00

'Bring channels into service
00 12 00 0a 00 00 ff 01 02 0d 03 00 08 00 0d 03 00 08 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 09 00 0d 03 00 09 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0a 00 0d 03 00 0a 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0b 00 0d 03 00 0b 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0c 00 0d 03 00 0c 17
   f0
```

```
00 12 00 0a 00 00 ff 01 02 0d 03 00 0d 00 0d 03 00 0d 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0e 00 0d 03 00 0e 17
   f0
00 12 00 0a 00 00 ff 01 02 0d 03 00 0f 00 0d 03 00 0f 17
   f0

'Virtual Span Control
'Tell the switch the spans are alive
00 0e 00 e2 00 00 FF 00 01 0C 02 00 08 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 09 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0a 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0b 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0c 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0d 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0e 00 01 00
00 0e 00 e2 00 00 FF 00 01 0C 02 00 0f 00 01 00
```

Host-Controlled Continuity

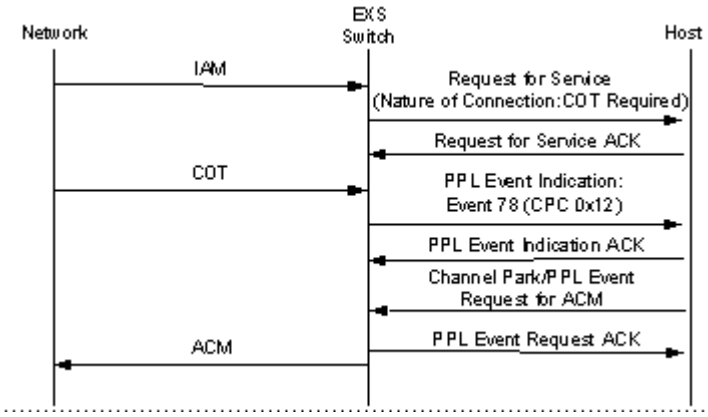
Since virtual T-ONE and E-ONE/J-ONE cards function the same way real cards would in the CSP, the switch will automatically attempt a Continuity Check for a CIC when an IAM is received. The Continuity Check is performed for the voice circuits (CICs). This involves testing whether a tone has been properly transmitted at one end and been received at the other on the CIC for which the call is intended. If the Continuity Check passes, then the call continues. If it does not, the call is released.

However, since no actual span exists, this Continuity Check will fail. To prevent this, the automatic Continuity Check is disabled in the switch. In its place Dialogic uses the “Host-controlled Continuity”. Using this feature, the host can initiate a Continuity Check manually by choice only.

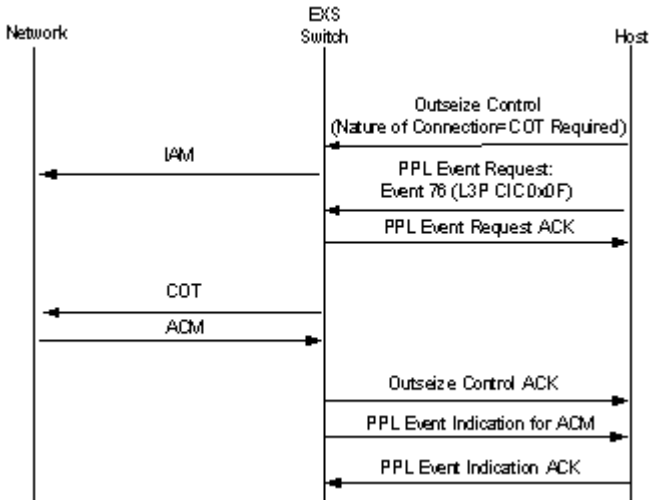
Call Flows for Virtual Spans

This section includes call flow examples for Virtual Spans in both ITU and ANSI.

Incoming Call: IAM Received with COT Required

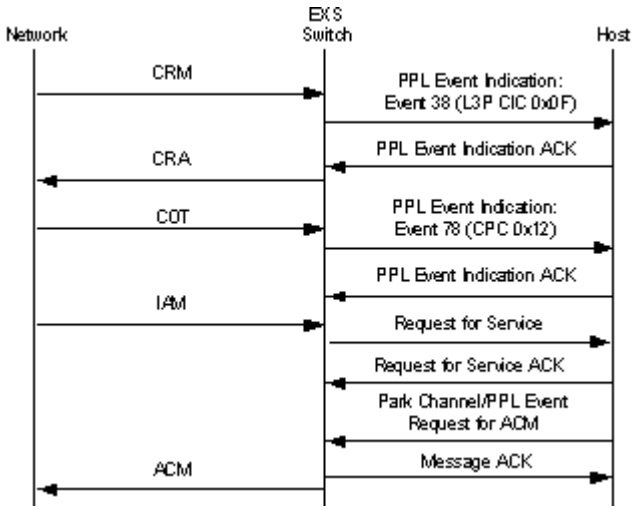


Outgoing Call: IAM Sent with COT Required

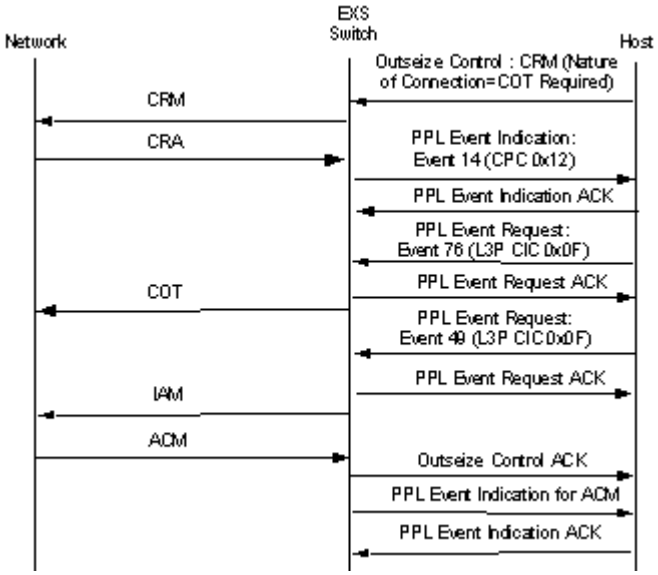


ANSI Call Flows This section includes call flow examples for Virtual Spans in ANSI only.

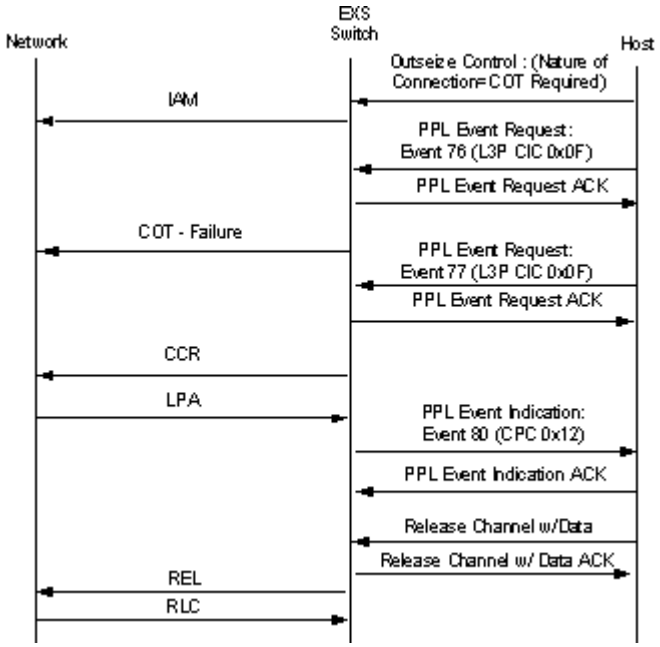
ANSI CRM Received: NOC=COT Required



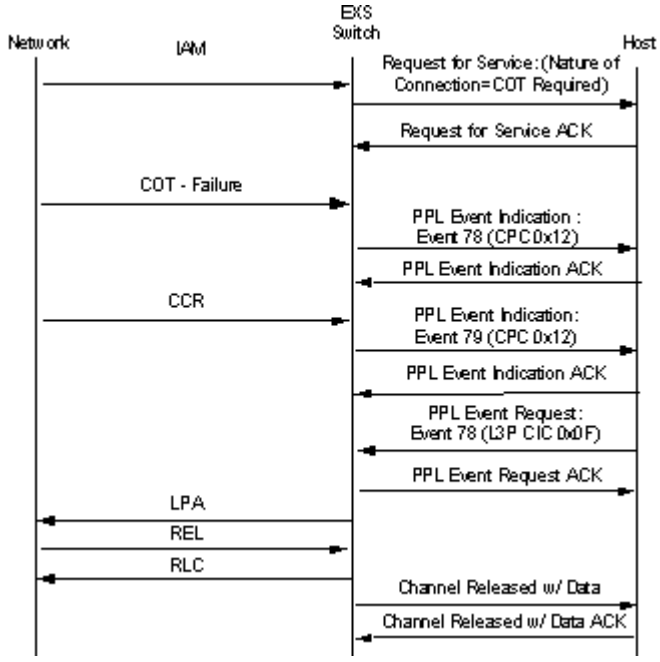
ANSI CRM Sent: NOC=COT Required



ANSI IAM Sent: COT Failure: Continuity Recheck



ANSI IAM Received: COT Failure: Continuity Re-check



Configuring the SS7 Multi-Protocol I/O

Purpose

This section describes how to configure the SS7 Multi-Protocol I/O (SS7 MP I/O) card which supports the V.35 electrical interface. The V.35 interface allows an SS7 link to terminate on a separate channel freeing the channel used on a span.

Before you begin

The system software treats the SS7 Multi-Protocol I/O card as another line interface card. When the system software detects the card (either when the system powers up or if the SS7 MP I/O is inserted while the CSP is running), it notifies the host using the *Card Status Report* message (0xA6).

Channels and Connectors

Up to four 2090 channels can be connected to a V.35 interface using the SS7 MP I/O card. The choice of electrical interface is set using the *SS7 Signaling Link Configure* message. Each port can be independently configured to run at 48, 56, or 64 Kbps. As with the electrical interface, the choice of data rate is set with the *SS7 Signaling Link Configure* message.

There are four timeslots available, one per port or connector. The timeslots map to channels as follows:

Timeslot	Channel
0	0
1	1
2	2
3	3

The SS7 MP I/O supports four channels, out of a possible 2048 in a 2090 chassis. This card differs from regular E1/T1 cards in that the system selects the four channels when the user assigns a Logical Span ID to the card. The system tries to allocate the four channels for an SS7 MP I/O card by first selecting unused channels on a T1 card. If these are not available, (that is, there are no T1 cards in the system), it then utilizes the full bandwidth required for an 8-span E1 card (256 channels), even though only four channels are used. However, up to four SS7 MP I/O cards will share this 256-channel block.

If there is no bandwidth available (that is, if 64 E1 spans are in the chassis), and the Assign Logical Span message is sent, the Assign Logical Span will receive a response of NACK with a status code of "0x28: No DSP Resources Available".

Configuring Multi-Protocol

The SS7 MP I/O card functions like the E1 or T1 card and is configured the same way. The following steps summarize the basic procedure.

Step	Action
1	Assign the Logical Span ID.
2	Bring span into service.
3	Configure the SS7 Signaling Stack as usual.
4	Bring SS7 links into service.

The I/O card is programmed for the link data rate and the electrical interface via the *SS7 Signaling Link Configure* and *SS7 Signaling Link Query* messages.

Sample Configuration File

Below is an example of a configuration file using the API format to configure an SS7 MP I/O card for V.35 traffic and to bring V.35 links into service.

```
' SS7 Software license
00 19 00 79 00 00 ff 01 02 24 10 46 46 41 59 34 4e 46 4b
   4d 42 41 4c 4c 32 30 4b
' Assign logical spans
00 0d 00 a8 00 00 ff 00 01 11 04 ff ff ff ff ' de-assign
   them all first
00 0d 00 a8 00 00 ff 00 01 11 04 00 14 36 00 ' V.35 Card,
   slot 0x36

------------------------------------------------------------------------

 Service State Configure, all spans in-service

 ------------------------------------------------------------------------

00 0d 00 0a 00 00 ff 00 01 0c 02 00 14 f0 00    'V.35
' SS7 Signaling Stack Configure with stack 00

'00 16 00 5c 00 00 ff 'header
'00 01 21 02 03 'Slot AIB
'00 'Stack
'00 00 00 06 ' OPC
```

```
'03 01 01 02 01 03 01 ' modules, type, variant, type,
   variant, type, variant
00 16 00 5c 00 00 ff 00 01 21 02 03 00 00 00 00 06 03 01
   01 02 01 03 01

Trace for stack 01:
'00 16 00 5c 00 00 ff 'header
'00 01 21 02 03 'Slot AIB
'01 'Stack
'00 00 00 03 'OPC
'03 01 01 02 01 03 01 modules, type, variant, type,
   variant, type, variant
00 16 00 5c 00 00 ff 00 01 21 02 03 01 00 00 00 03 03 01
   01 02 01 03 01


' PPL Timer Configure (T20=10msec)
 00 10 00 cf 00 00 ff 00 01 08 01 00 00 3d 01 01 00 01
 00 10 00 cf 00 00 ff 00 01 08 01 01 00 3d 01 01 00 01


' PPL Timer Configure (T21=20msec)
 00 10 00 cf 00 00 ff 00 01 08 01 00 00 40 01 01 00 02
 00 10 00 cf 00 00 ff 00 01 08 01 01 00 40 01 01 00 02


' SS7 Signaling Link Set Configure for Stack 00
'00 0f 00 5d 00 00 ff ' Header
'00 01 1e 02 00 00 'Link Set AIB
'00 00 00 03 'APC
00 0f 00 5d 00 00 ff 00 01 1e 02 00 00 00 00 00 03

' SS7 Signaling Link Set Configure for Stack 01
'00 0f 00 5d 00 00 ff 'header
'00 01 1e 02 01 01 'Link Set AIB
'00 00 00 06 'APC
00 0f 00 5d 00 00 ff 00 01 1e 02 01 01 00 00 00 06


' SS7 Signaling Link Configure for stack 00
'00 14 00 5e 00 00 ff 'Header
'00 02 1f 03 00 00 04 0d 03 00 14 00 'combination Link ID
   'and Channel AIB
'02 'signaling link code
'00 'data rate
'01 'V.35 electrical interface, dte
00 14 00 5e 00 00 ff 00 02 1f 03 00 00 04 0d 03 00 14 00
   02 00 01

' SS7 Signaling Link Configure for stack 01
```

```
'00 14 00 5e 00 00 ff 'Header
'00 02 1f 03 01 01 05 0d 03 00 14 01 'combination Link ID
   'and Channel AIB
'02 'signaling link code
'00 'data rate
'02 'V.35 electrical interface, dce
00 14 00 5e 00 00 ff 00 02 1f 03 01 01 05 0d 03 00 14 01
   02 00 02

' SS7 Signaling Route Configure for stack 00
'00 14 00 5f 00 00 ff ' Header
'00 01 20 05 00 00 00 00 00 'SS7 Destination/Route AIB
'00 00 00 03 'DPC
'00 'Link Set ID
'00 'Priority
00 14 00 5f 00 00 ff 00 01 20 05 00 00 00 00 00 00 00 00
   03 00 00

' SS7 Signaling Route Configure for stack 01
'00 14 00 5f 00 00 ff ' Header
'00 01 20 05 01 00 01 00 01 'SS7 Destination/Route AIB
'00 00 00 06 'DPC
'01 'Link Set ID
'00 'Priority
00 14 00 5f 00 00 ff 00 01 20 05 01 00 01 00 01 00 00 00
   06 01 00

' Service State Configure (SS7 links for stack 00)
'00 0d 00 0a 00 00 ff 'Header
'00 01 09 02 00 04 'SS7 Signaling Link AIB
'f0 'Action
'00 'Flag
00 0d 00 0a 00 00 ff 00 01 09 02 00 04 f0 00 ' V.35

'00 0d 00 0a 00 00 ff 'Header
'00 01 09 02 00 05 'SS7 Signaling Link AIB
'f0 'Action
'00 'Flag

00 0d 00 0a 00 00 ff 00 01 09 02 01 05 f0 00
```

Clock Source

You must use the *SS7 Signaling Link Configure* message to set the clock source for this card. The clock source can be specified independently for every port. The options for clock source include the following:

Data Terminal Equipment (DTE): Derives clock from incoming data.

Data Communications Equipment (DCE): Derives clock from CSP system timing.

Configuring JT ISUP Variant Stack

Purpose

The present ISUP implementation in the system software, by default, supports Japanese Telecom ISUP variants in the SS7 stack. If ISUP messages must be added or removed to conform to JT ISUP variants, use *SS7 ISUP Message Format Configure*.

Before you begin

Obtain an ISUP license from your sales representative.

Configuring JT ISUP Variant Stack

The steps below describes how to configure the JT ISUP variant stack.

1 Download your ISUP license using *Product License Download.*

2 Create a stack for the JT variant using *SS7 Signaling Stack Configure*.

- Set the MTP Module Type (0x01) to the JT Module Variant (0x08)
- Set the ISUP Module Type (0x02) to the JT Module Variant (0x08)
- Set the L3P Module Type (0x03) to the JT Module Variant (0x08)

3 Download the customized PPL tables for the variant. Note that the JT/TTC variant does not require a protocol, only that an SS7 stack be configured. The NTT/DDI variant requires customized PPL tables. Contact Technical Support for these tables.

4 Set the configuration byte for the PPL components, HMDT and HMRT to a variant, 0x01. Modify PPL Config Bytes with the *PPL Configure* message, as follows:

- HMDT (0x002B)

  Set Config Byte 0x01 (Variant) to 0x02 (Japan/TTC) or 0x04 for NTT

  Set Config Byte 0x04 (National Indicator) to 0x00 (International)

  Set Config Byte 0x12 (Japan A/B plane verification) to enable Japan A/B plane verification.

- HMRT (0x002C)

  Set Config Byte 0x01 (Variant) to 0x02 (Japan/TTC) or 0x04 for NTT

  Set Config Byte 0x04 (National Indicator) to 0x00 (International)

  Set Config Byte 0x06 (Message Priority) to 0x03.

  Set Config Byte 0x12 (Japan A/B plane insertion) to enable Japan A/B plane insertion.

5 Modify PPL Timer values with the *PPL Timer Configure* message, as follows:

- TPRC (0x003D)

  Set timer 1 (T20) to 0x0001 (10ms)

- TSRC (0x0040)

  Set timer 2 (T21) to 0x0002 (20ms)

Important! MTP PPL configuration byte and PPL timer values are listed in the chapter, *SS7 PPL Information*. National variants may require additional User Part and protocol modifications.

END OF STEPS

Combined Link Set

Overview

The Combined Link Set feature is part of the CSP architecture's MTP3 layer and supports ANSI and ITU applications. This feature provides the following options:

- The host can configure a combined link set as part of the route configuration, using the *SS7 Signaling Route Configure* (0x5F) message.
- Load sharing of traffic is possible within the combined link set.
- The 5-bit Signaling Link Selection (SLS) is the default value for the configuration byte (0x0E) in the MTP3 HMRT PPL component (0x2C):

  The 5-bit SLS is the default for hierarchical routing to enable backward compatibility with traffic using 32 SLS values to assign to links in a network.

  The 8-bit SLS option is not supported in this release.

Combined Link Set

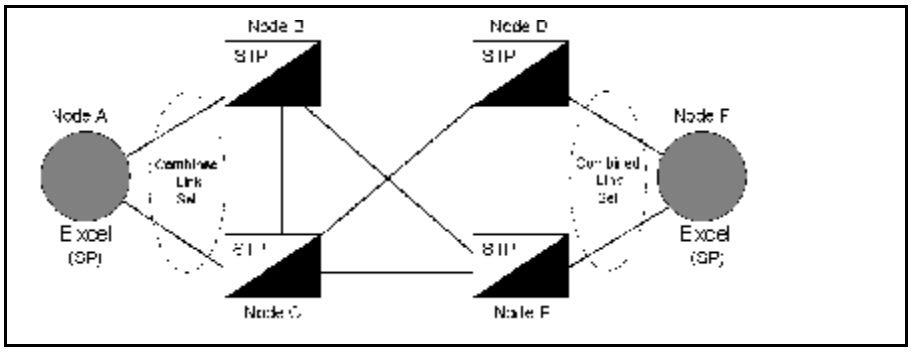
A route is the combination of one or more link sets used to reach a specific destination. Each link set can belong to more than one route, and contains from 1 to 16 links. A primary route exists for every destination, and represents the fastest path, and least cost, to that destination.

A combined link set establishes the availability of alternate routing configurations that can each serve as a primary path, providing load sharing across the network. Within a combined link set, which can maintain up to a total of 32 links in any combination of link sets, every link is given an equal priority in the routing assignments to the same destination. This means that the more paths that exist to a specific destination in a network, the more reliably messages can be transmitted, even when one or more configured routes fail or experience power outage for any reason.

In the combined link set feature, a routing table tracks the links that have been configured, lists the links that are currently available, and indicates when an update to the network topology is required.

The combined link set will contain links to both Signal Transfer Points (STPs) in an SS7 network that are paired for redundancy and connected to each other by cross links). Refer to the figure below:

Figure 2-14 Combined Link Set Topology



The SLS is the Least Significant Byte (LSB) of the CIC Number, which is found in the routing label of the user part assigned.

If there is only one link set, the LSB serves to identify the link, but the Most Significant Byte (MSB) is not used.

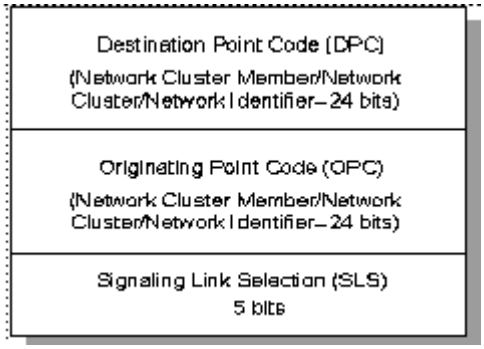
See *SS7 Signaling Route Configure* (0x5F) in the *API Reference* for selection of the MSB or LSB values which allow the route to be configured as part of a combined link set.

Routing Label

SS7 addressing for nodes, networks, and groups of signaling points in a specific area is done through the use of a point code. A unique 24-bit (three-octet) point code or numeric address is part of the routing label used in the ANSI SS7 MTP3 layer to route the signaling information for a call. This label, which is part of the Signaling Information Field (SIF) of a Message Signal Unity (MSU), contains:

- An 8-bit Destination Point Code (DPC) identifying the termination point
- An 8-bit Originating Point Code (OPC) identifying point of origin for the message.
- A Signaling Link Selection field (SLS) that is used to distribute message traffic over all possible redundant links and routes.

Figure 2-15 Routing Label



Other information about the call, which can include data such as the connection type (bearer capability), carrier identification, and the identification of an unlisted calling number, is also sent as part of the routing label. For the purposes of this document, only the 3-octet point code is relevant.

Destination Point Code

When the routing function is activated, the DPC is used by a node to select and determine the link set or combined link set for an outgoing message. The network identifier (Network ID) in the DPC directs the message to a particular SS7 network.

The DPC may contain the point code of the next node involved in the transmission route or it can be the final node for the entire route.

Signaling Link Selection

Signaling Link Selection (SLS) is used for load sharing when two or more links connect adjacent signaling points. Its function is to distribute the traffic evenly. Each signaling link is assigned an SLS value. This SLS value never changes and identifies the link that the message is routed over. The SLS value must match at both ends of the signaling link.

The SLS is used to select the link within a selected link set to:

- Ensure the sequencing of messages. Any two messages that are sent with the same SLS will always arrive at the destination in the same order in which they were originally sent. That is, if messages are intended to be kept in sequence, the same SLS code will be used so that all such messages take the same path to the destination. For example: a signaling link with ISUP will have the same SLS code used for all messages related to that particular link.

The purpose of the above functionality is to provide the option to reconfigure the signaling network if there are signaling link or signaling point failures, and to control traffic in the event of congestion,

blocks or bottlenecks. When a failure occurs, the reconfigurations are carried out so that messages are not lost, duplicated, or placed out of sequence, and excessive message delays are avoided.

The figures below display the SLS values both for a single link set containing four individual links and a combined link set containing two link sets:

Figure 2-16 Load Sharing within a Single Link Set

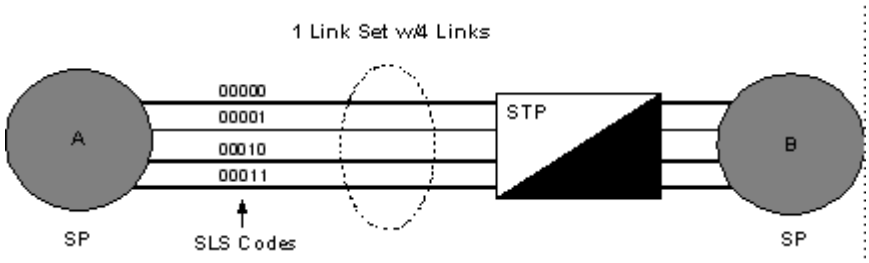
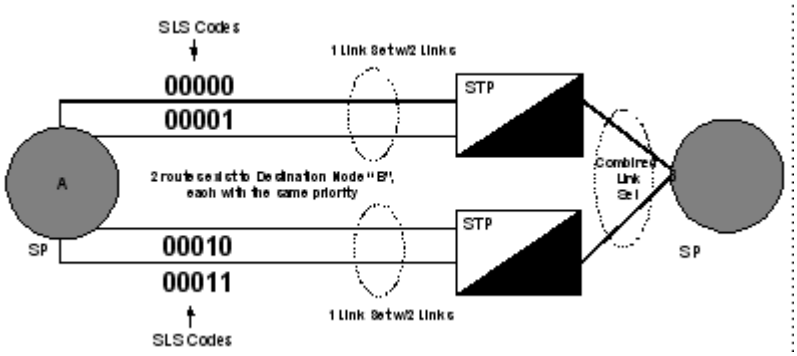


Figure 2-17 Load Sharing in a Combined Link Set



Implementing Combined Link Set

To implement the Combined link set with the 5-bit SLS routing table, the following MTP3 PPL components apply:

TSRC

TSRC handles the traffic management route control of MTP3; currently it maintains a routing table per destination, and the routing data is kept in the SLS table. This table is updated when a route is added or removed, or when the first MSU is routed for that route.

HMRT

The SLS table is updated in HMRT to include new routes and all the links of the combined link set, in order to distribute the traffic evenly.

TCOC and TCBC

These components are used for the routing functions of changeover and changeback. It is in these components that all necessary changes are made to distribute the traffic based on the normal SLS table. The TSRC component marks the SLS table whenever a new route is added or deleted. This means the table must be updated. Then when an MSU is routed to that destination and the SLS table is already marked with the addition or deletion action, the table will be updated.

All three components--TCBC, TCOC and TSRC--will update the list of available links when required.

Changeover

When changeover to another route begins, traffic is buffered for that link. All destinations will be marked when that is complete. Then traffic will be sent on alternate links once the changeover ends, and the SLS table will be updated to indicate the new links that are now carrying traffic. Again, this update will only take place when the first MSU is routed.

Changeover diverts traffic from an unavailable link to one or more available links, without message loss, mis-sequencing or duplication.

Important! Traffic is load shared between the two links from the SP to the mated pair of STPs. The STP monitors the error rate. If a link's functionality deteriorates, the STP will initiate a changeover, thus routing all traffic on to one link. When the faulty link is repaired, the STP will then order a changeback request and traffic is routed back to that link and load sharing is restored.

Changeback All destinations are marked when changeback begins, and traffic is re-distributed from alternate links to the link coming up. The SLS table is updated immediately and the alternate links start to buffer traffic. This update occurs, however, only if there is an MSU to be routed to the destination.

Once Changeback is acknowledged by the link coming up, it will buffer the traffic for that alternate link, according to the updated SLS table.

Changeback diverts traffic back to the original route once it becomes available.

Important! A changeover order is always sent over an alternate link, while a changeback order can be routed over any available link.

Dual Ethernet Port for SS7 Series 3 Card

Overview

With the implementation of M3UA software, Dialogic recommends that both Ethernet physical ports on the CCS I/O Series 3 card are utilized as follows:

- Port A - Host to SS7 card traffic
- Port B - M3UA traffic to the network.

Configuring Ethernet Ports

This section describes how to configure both Ethernet ports.

If you configure Port A to be on one subnet and Port B to on a different subnet, messages will not come from the desired physical ports.

Follow the steps below to configure the Ethernet ports:

1. From the *IP Address Configure* message (0x00E7), include the Module, IP Address, and Subnet Mask TLV (0x01).
2. Include the following module values and corresponding IP Addresses and Subnet masks:

   0xFF - Port A
   0x01 - Port B

3. Include a Reset TLV at the end of this message to force the SS7 Series 3 card to reset. This step allows the changes to get applied to the card.

Configuring the second Ethernet port is optional but if you enter an IP Address for Ethernet port B you must enter an associated Subnet Mask.

Important! Although a gateway IP Address may be configured for both Port A and Port B, you only supply one gateway address per physical interface.

Message Transfer Part (MTP)

Overview

This section includes information on the following MTP features:

- Preventive Cyclic Retransmission (PCR)
- Craft Alerting
- ANSI Activation Link Test
- Periodic Link Test
- National Variant Support for China
- National Variant Support for Japan TTC and Japan NTT/DDI See also *MTP3 for Japanese Telecommunications (2-101)* and *Atomic Function 98 (2-124)*
- Link Set Configuration

Preventive Cyclic Retransmission (PCR)

Preventive Cyclic Retransmission (PCR) is an alternative method for SS7 signaling link error correction. PCR should be used in situations with large propagation delays, such as satellite circuits.

MSUs are stored by the transmitting terminal until a positive acknowledgment (ACK) is received. When no new MSUs are to be sent, unacknowledged MSUs are retransmitted cyclically until positively acknowledged.

PCR is enabled and configured with the PPL Config Bytes of the MTP2 Transmission Control (TXC) component (0x26.)

Configuration

To implement PCR, you must modify the Config Bytes of the MTP2 TXC component (0x26) with the *PPL Configure* message as follows:

- Change the value of Config Byte 2 to 0x01 (PCR) using the *PPL Configure* message.

MTP PPL Config Byte values are listed in the *SS7 PPL Information*.

- Set the N2 Parameter Value (Config Bytes 4 and 5) to an appropriate value for the signaling link loop delay using the following formula:

N2 = Tloop (in microseconds)/125 microseconds +1
For example, the typical satellite loop delay is approximately 500 milliseconds, therefore:
N2 = (500000/125) +1 = 4001 bytes

The default value for Config Bytes 4 and 5 is 0xFFFF, which is for Basic mode error checking.

Important! To initiate configuration changes, take the link out of service and then bring it back in service with the *Service State Configure* message. Configuration changes will also take affect if the link fails and realigns.

Example

This example *PPL Configure* message shows the typical PPL Configuration required to implement PCR. The following configuration is performed:

- Config Byte 2 (Mode) is set to 0x01 (PCR)
- Config Bytes 4 and 5 (N2 Parameter Value) are set to 0x0FA1 (4,001 bytes)

Trace

```
H->X FE 00 15 00 D7 00 00 FF 00 01 09 02 03 00 26 01 03 02
   01 04 0F 05 A1 CS
```

BYTE	Field Description	Value
0	Frame	0xFE
1	Length, MSB	0x00
2	Length, LSB	0x15
3	Message Type, MSB	0x00
4	Message Type, LSB	0xD7
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0xFF
8	AIB (starting with Byte 0): Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element 1: Originating Channel Address Type	0x09 (SS7 Signaling Link)
11	Data Length	0x02
12	Data[0] Stack ID	0x00

BYTE	Field Description	Value
13	Data[1] Link ID	0x01
14	PPL Component ID, MSB	0x00
15	PPL Component ID, LSB	0x26 (SS7 MTP2 TXC)
16	PPL Entity	0x01 (PPL Config Bytes)
17	Configuration Data Number of Data Locations	0x03
18	Location 1: Byte Number	0x02
19	Location 1: Data	0x01 (PCR)
20	Location 2: Byte Number	0x04
21	Location 2: Data	0x0F
22	Location 3: Byte Number	0x05
23	Location 3: Data	0xA1 (with byte 4: 4,001)
24	Checksum	0xCS

Craft Alerting

If Craft Alerting is enabled, the host is sent a *PPL Event Indication* message with an event of Link Activation Failure (0x02) if the CSP experiences persistent problems while attempting to align SS7 signaling links.

By default, Craft Alerting is enabled for ANSI and disabled for ITU. Craft Alerting is enabled (0) or disabled (1) with PPL Config Byte 1 of the MTP3 LSAC component (0x002E.) The persistent timer (TA) is specified by PPL Timer 1 of the LSAC component. The default is 60 s.

Craft Alerting is not supported by the Japanese variants.

MTP3 Signaling Link Tests

The testing of signaling links is provided through the LSAC (Signaling Link Activity Control) PPL component (0x002E). See ANSI T1.111 for more information about MTP3.

Activation Link Test

If the Activation Link Test is enabled, an SS7 link is not brought in service until the point codes on either end of the link are verified by the CSP. If the point codes are not verified, the host is sent a *PPL Event Indication* of Signaling Link Test Failure (0x03).

The Activation Link Test is enabled by default for all variants. It can be disabled with PPL Config Byte 2 of the MTP3 LSAC component.

The Activation Link Test cannot be disabled for the Japanese variants.

Periodic Link Test

If the Periodic Link Test is enabled, the CSP periodically verifies the point codes and the Signaling Link Test on the link. If the point codes and the Signaling Link Code (SLC) do not verify, the link fails and the host is sent a *PPL Event Indication* of Signaling Link Test Failure (0x03).

The Periodic Link Test is enabled by default for all variants. It can be disabled with PPL Config Byte 3 of the MTP3 LSAC component. The default period of the test is 60 seconds. This is configured with PPL Timer 3 of the MTP3 LSAC component.

National Variant Support for China

To enable MTP support for the China variant, perform the following:

- Configure the signaling stack for the ITU variant

  With the *SS7 Signaling Stack Configure* message, configure the Module Variant for ITU (0x01)

- Modify PPL Config Bytes with the *PPL Configure* message, as follows:

  HMDT component (0x002B)

  - Set Config Byte 1 (Variant) to 0x03 (China)

  HMRT component (0x002C)

  - Set Config Byte 1 (Variant) to 0x03 (China)

  Important! MTP PPL configuration byte values are listed in the chapter, *SS7 PPL Information*. National variants may require additional User Part and protocol modifications.

Link Set Configuration

“A” link sets must be configured with even numbered Link Set IDs (for example; 0, 2, 4.) “B” links must be configured with odd Link Set IDs (for example; 1, 3, 5.) An A/B link set pair configured to an adjacent signaling point must be configured with consecutive Link Set IDs (for example; 0/1, 2/3, 4/5.)

MTP3 for Japanese Telecommunications

Overview

Dialogic's system software SS7 stack has been enhanced to support MTP3 variants used in Japan: JT, NTT and DDI. While the term JT is used as a variant within the system software, JT is also the TTC standard for MTP3 used in Japanese telecommunications networks. The JT variant supports Japan Telecom. The NTT variant supports Nippon Telegraph and Telephone of Japan. The DDI variant supports DDI Corporation of Japan.

MTP3 JT, NTT, DDI Variants

The system software SS7 stack supports MTP3 variants: JT, NTT and DDI, which support networks in Japan. The Japanese telecommunications specification, TTC JT-Q707, defines Signal Routing Test (SRT) to verify a signaling route for end-to-end communication. The telecommunications specification of the International Telecommunication Union (ITU-T), ITU-T Q.707, defines Signaling Link Test (SLT), but does not define SRT. Even though JT-Q707 does not define SLT, the software allows a host to locate a destination point code (DPC) and test a signaling route to a switch using JT standards.

Hardware Requirements

The feature requires an SS7 Series 3 and SS7 I/O Series 3 Card Set.

Locating a DPC and Testing a Signaling Route

The MTP3 component, Signaling Link Test Control (0x41) allows a switch to locate a DPC and test a signaling route. With JT, NTT, and DDI variants, SLTC is not performed by default. You must manually activate this component by sending a *PPL Event Request* message. The table below describes the process of testing a signaling route and accessing a destination using the SLTC component.

Stage	Description
1	To initiate a Signaling Route Test, the host sends a PPL Event Request (0x44) to the MTP3 PPL component, SLTC.
2	If the PPL Event Request is successful in locating a DPC, the switch immediately sends an ACK (acknowledgment) to the host. If the switch is unsuccessful in detecting a DPC a NACK (Negative Acknowledgment) is sent to the host and the signaling route test is not initiated.
3	If the PPL Event Request was successful, a Signaling Route Testing message is sent to the remote DPC.

Stage	Description
4	After the test is completed, the switch sends a PPL Event Indication (0x43) to the host with the results of the test.

Example Configuration File

This section shows examples of configuration files.

PPL Event Request: To Initiate SRTC

The script below uses the PPL Event Request message to activate a route signaling test.

00 1d 00 44 00 0d ff\
00 01 09 02 00 00 \ ' AIB type - Stack/Link(0x09)
00 41 00 01 \ ' PPL comp/Event ID (0x41/0x0001)
01 02 1e 0a 00 01 00 12 00 04 00 00 02 22 ' ICB sub type - 0x1E

PPL Event Indication - Result of SRTC

The following is an example of a configuration script file using the PPL Event Indication message to get the results of a signaling route test.

00 23 00 43 00 00 00 \
00 01 09 02 01 04 \ ' AIB type - Stack/Link(0x09)
00 41 00 01 \ ' PPL comp/Event ID (0x41/0x0001)
01 02 1E 10 00 02 00 11 00 02 00 10 00 12 00 04 00 00 01 11 ' ICB sub type - 0x1E

MTP3-to-Host

Introduction This section explains MTP3-to-Host feature including:

- *MTP3-to-Host Functionality (2-104)*
- *Software Modifications (2-110)*
- *Configuring MTP3-to-Host (2-114)*
- *MTP3-to-Host Example (2-115)*

MTP3-to-Host Functionality

Introduction

The MTP3-to-Host feature allows the option for the SS7 User Parts to be resident in the host or on the local SS7 Series 3 card. (This feature is supported only via the SS7 Series 3 card.)

With host-resident User Parts, information is forwarded to the host with MTP traffic handled by the SS7 Series 3 card on the CSP.

The host destination can be one of many destinations. You configure it using the *PPL Configure* (0x00D7) message.

The MTP3 layer checks if a remote User Part is active and sends the message to the appropriate remote destination.

Based on Remote User Part configuration, the signaling message will bypass the CSP internal User Part and be delivered to the remote host via the Matrix host connection (that is, the Matrix Controller I/O card).

The MTP3-to-Host feature may co-exist with other modules on the same stack. For example, you can configure a stack consisting of MTP, ISUP, L3P, SCCP, and TCAP modules. MTP, L3P, SCCP, and TCAP will be resident on the SS7 Series 3 card, whereas the HMDT configuration byte setting redirects ISUP traffic to a remote host.

Prerequisites

- This feature assumes that there is a workable User Part available in a selected remote host.
- You first configure a stack and then redirect the User Part information to the remote host.
- You have to license this feature (Key Type 0x3134 in Product License 0x24 ICB).

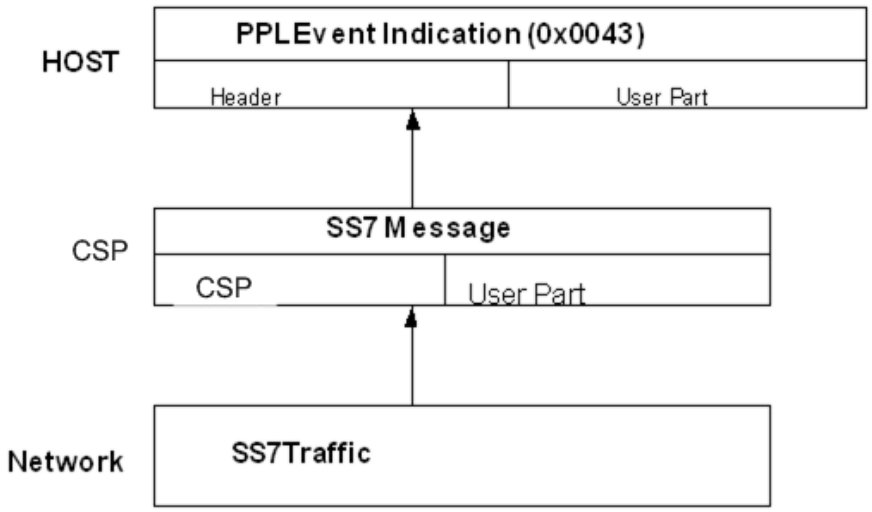
Data User Part

The Data User Part (DUP) is a call control user part designated for switched data services.

Message Structure

The task of sending resident user parts to a remote host is basically performed through MTP3 detecting user parts in SS7 messages and forwarding those user parts to the appropriate host in a *PPL Event Indication* (0x0043) message as indicated in *Figure 2-18, User Part in PPL Event Indication Message (2-105)*:

Figure 2-18 User Part in PPL Event Indication Message



MTP3 Components

Most of the changes required to support the MTP3-to-Host functionality reside on the CSP itself. This functionality is made possible with the following MTP3 components:

- HMDT - Message Distribution

HMDT module is a component of MTP3 layer. This component routes the TRANSFER indications from MTP2 to the appropriate User Part.

- HMRT - Message Routing

HMRT is a component of the MTP3 layer. This component routes messages coming from layer 4 and selects the signaling link over which it is routed. It handles TRANSFER requests from layer 4.

- TSFC - Traffic Signaling Flow Control

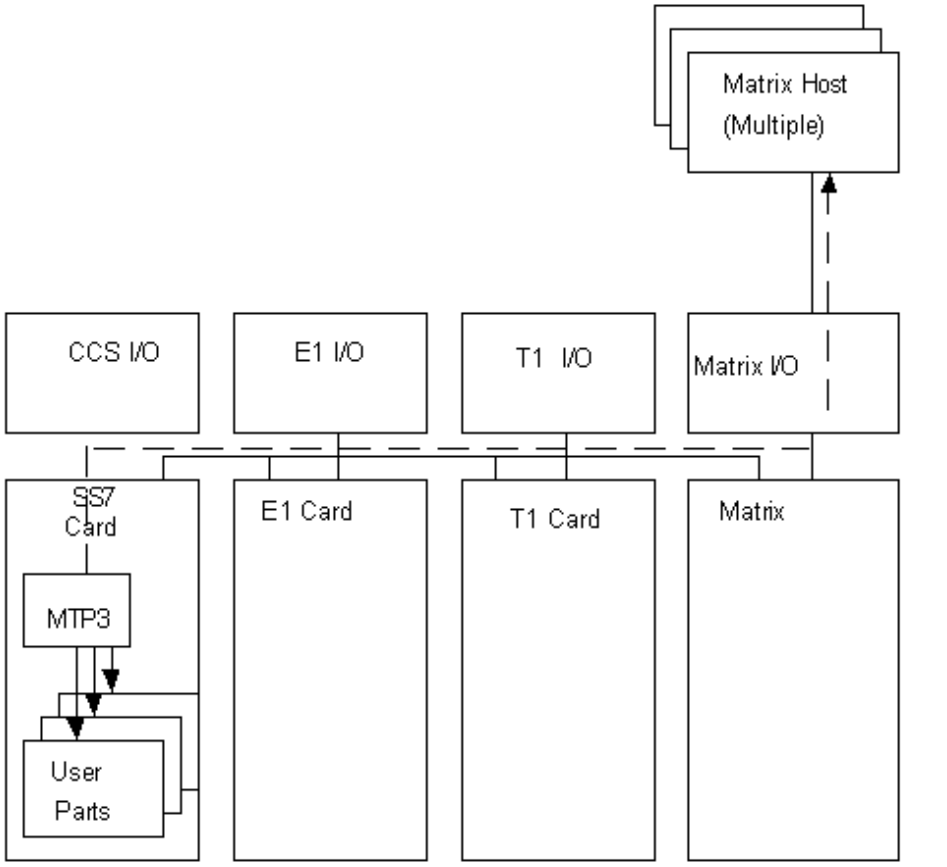
TSFC Module is a component of the MTP3 layer. This component controls the signaling traffic flow in the case when the signaling network is not capable of transferring all signaling traffic because of network failure or congestion.

When an SS7 user part is activated for a remote host then the destination of the message can be any one of several remote hosts. This option is configurable using the PPL Configure message.

Data Flow

The figure below shows the data flow between MTP3 components and the remote host.

Figure 2-19 Data Flow between MTP3 Components and Host



The HMDT component of MTP3 will use new configuration bytes to allow you to program which SS7 user parts will remain local to the CSP and which user parts will reside in a remote host. The default is that all the user parts reside on the CSP.

MTP3 HMDT (0x2B) configuration bytes 0x27-0x33 are used to indicate the port number of the remote host for the corresponding user part. For example, if user part in configuration byte 0x19 is set to 0x02 (ISUP remote host), then configuration byte 0x29 will give the port number of the ISUP remote host. The default value of these bytes is 0xFF.

The new PPL event indication from the HMDT component to the host reports TRANSFER IND and sends the corresponding data to the host.

The new PPL event indications from the component TSFC to Host report the STATUS of a DPC or PAUSE and RESUME indications for an affected DPC.

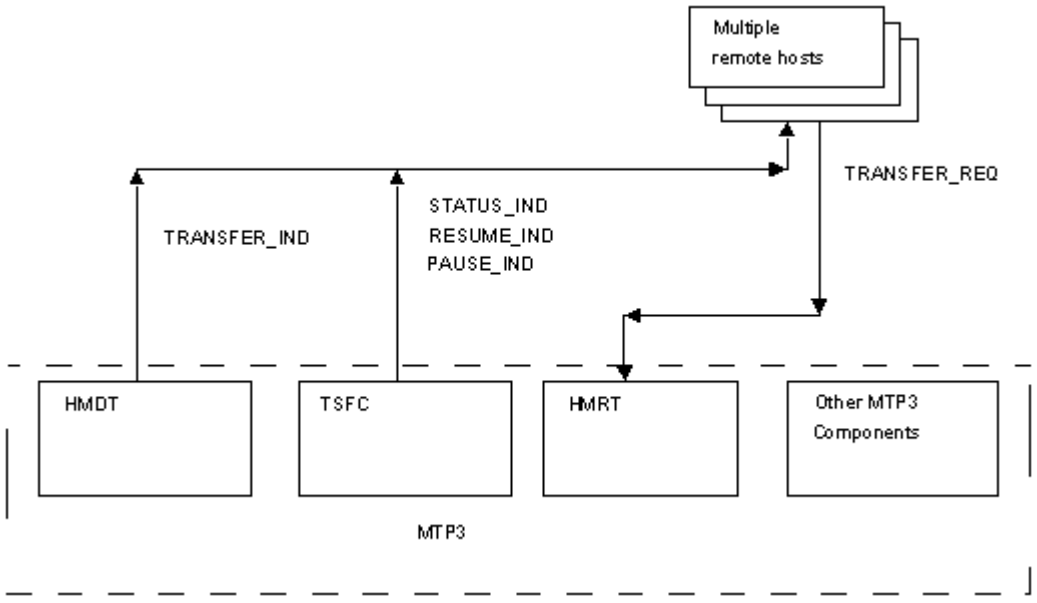
The new PPL event request from the host to HMRT component is a TRANSFER Request from the Host to the HMRT component.

The default sequence of these events is from the corresponding MTP3 component to a specific SS7 user part on the CSP. However, if you configure the HMDT configuration bytes then the events flow to a remote User Part resident in a remote host. This may be achieved on a per User Part basis. Also, there may be multiple remote hosts. In this case, the port number of the remote host is stored in a specific HMDT configuration byte.

Data Flow

Figure 2-20 shows the data flow between MTP3 components and the remote host. The dashed line represent MTP3 components in the CSP.

Figure 2-20 Data Flow Between Components and Host



The HMDT component of MTP3 will use new configuration bytes to allow you to program which SS7 user parts will remain local to the CSP and which user parts will reside in a remote host. The default is that all the user parts reside on the CSP.

Configuration Bytes 0x24-0x26 are used to indicate the port number of the remote host for the corresponding user part. For example, if user part in Configuration Byte 0x19 is set to 0x02 (remote host), then Configuration Byte 0x24 will give the port number of the remote host. The default value of these bytes is 0xFF.

The new PPL event indication from the HMDT component to the host reports TRANSFER IND and sends the corresponding data to the host.

The new PPL event indications from the component TSFC to Host report the STATUS of a DPC or PAUSE and RESUME indications for an affected DPC.

The new PPL event request from the host to HMRT component is a TRANSFER Request from the Host to the HMRT component.

The default sequence of these events is from the corresponding MTP3 component to a specific SS7 user part on the CSP. However, if you configure the HMDT configuration bytes then the events flow to a

remote User Part resident in a remote host. This may be achieved on a per User Part basis. Also, there may be multiple remote hosts. In this case, the port number of the remote host is stored in a specific HMDT configuration byte.

Software Modifications

Purpose

The following summarizes the changes to the software to support this feature with links to the detailed formats.

HMDT PPL Information

Configuration Bytes

The HMDT (Message Distribution) component of MTP3 uses configuration bytes to allow you to program which SS7 user parts will remain local to the CSP and which user parts will reside in a remote host. The default is to have all the user parts in the CSP.

Refer to *MTP3 HMDT Configuration Bytes (7-183)*.

PPL Event Indications

There is a PPL Event Indication from the HMDT component to the host. The events reports TRANSFER IND and sends the corresponding data to the host.

Refer to *MTP3 HMDT PPL Event Indications (7-191)*.

TSFC PPL Information

PPL Event Indications

There are three PPL event indication from the component TSFC (Traffic Signaling Flow Control) to the host. These events report the STATUS of a DPC or Pause and RESUME indications for an affected DPC.

Refer to *MTP3 TSFC PPL Event Indications (7-208)*.

HMRT PPL Information

PPL Event Request

There is a PPL event request, TRANSFER, from the host to the HMRT (Message Routing) component.

The default sequence of these events is from the corresponding MTP3 component to a specific SS7 user part of the CSP. However, if you configure the HMDT configuration bytes, then the events flow to a remote user part resident in a remote host. You can achieve this process on a per user part basis. Also, there might be multiple remote hosts. In this case, the port number of the remote hot is stored in a specific HMDT configuration byte.

Refer to *MTP3 HMRT PPL Event Requests (AIB: Stack/Link 0x08) (7-200)*.

ICB

Below is the ICB for MTP to remote UP.

0x002B MTP3 User Part Parameter

Used in:
PPL Event Request message
PPL Event Indication message

ICB Type	0x03 (Extended Data)
ICB Subtype	0x002B MTP3 User Part Parameter
ICB Length	Length of parameters, in bytes
Data[0]	Parameter 1 Type ID
Data[1]	Length of parameter 1data in bytes
Data[2]	Parameter 1 Value Variable size
:	
Data[:]	Parameter n Type ID
Data[:]	Length of parameter n data in bytes
Data[:]	Parameter n Value Variable size

PPL Parameters

ICB Type	0x03 (Extended Data)
ICB ID	0x0424
Data Length 2 bytes	0x000D
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8-9]	CIC - Circuit ID Code
Data[10]	SI - Service Indicator
Data[11]	MP - Message Priority
Data[12]	NI - Network Indicator

ICB Type	0x03 (Extended Data)
ICB ID	0x0425
Data Length 2 bytes	0x000C
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8]	SI - Service Indicator
Data[9]	MP - Message Priority
Data[10]	NI - Network Indicator
Data[11]	SLS - Signaling Link Set

ICB Type	0x03 (Extended Data)
ICB ID	0x0426
Data Length 2 bytes	Up to 0x010E
Data[0-1]	Length - Length of MSU Data
Data[4-n]	MSU Data - Length up to 0x010C

ICB Type	0x03 (Extended Data)
ICB ID	0x0427
Data Length 2 bytes	0x0006
Data[0-3]	Affected DPC
Data[4]	Cause Value Value 0 - congestion level 0 Value 1 - congestion level 1 Value 2 - congestion level 2 Value 3 - congestion level 3 Value 4 - User Part Unavailable Value 5 - User Part Unequipped Value 6 - User Part Inaccessible Value 7 - Signaling Link Congestion level 0 Value 8 - Signaling Link Congestion level 1 Value 9: Signaling Link Congestion level 2 Value 10: Signaling Link Congestion level 3 Value 11: Signaling Link Congestion level 4
Data[5]	User Information Octet value

Configuring MTP3-to-Host

Download License

Before you can configure this feature, you must download the license as follows:

Send the *Product License Download* (0x0079) message with the Product License (0x24) ICB containing key type 0x3134. This key type enables the MTP3-to-Host feature.

Configure

Follow the steps below to configure the MTP3-to-Host feature.

1. Ensure that the SS7 stack is configured with local ISUP.
2. Send the *PPL Configure* (0x00D7) message to the CSP with configuration byte 0x19 set to 0x02 (Remote Host) on component 0x002B (MTP3 HMDT).

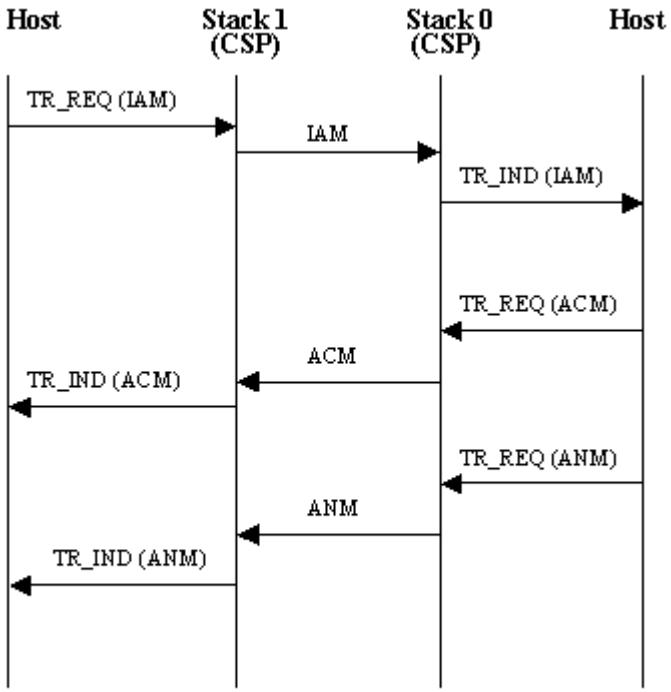
   Refer to *PPL Configuration Bytes for HMDT Component (0x002B)*.
3. (Optional Step). You can force the routing of the messages to a particular port. Configure byte 0x29 on component 0x002B.

   For example, a value of 0x02 forces the *PPL Transfer Indication* message to be sent to port 0x3144 (no indication received on any other port). The host can still send the Transfer Request messages from any port.

MTP3-to-Host Example

ISUP Call Flow

The following example illustrates the message flow from an ISUP call where both stacks are configured for remote ISUP. This call flow include the call establishment and release. Each step in the call flow is described following the two figures.



Call Flow Steps

The following steps break down the call flow. The OPCs of the stacks are as follows:

- Stack 0: OPC is 0-0-3
- Stack 1: OPC is 0-0-6

1. The host sends the Transfer Request (0x0A) for IAM to the Matrix Controller for Component MTP3 HMRT (0x002C).

```
H->X
```

```
00 44 00 44 00 28 04 00 01 08 01 01 00 2c 00 0a 01 03 00
   2b 00 30 04 25 00 0c 00 00 00 03 00 00 00 06 05 00 02
   00 04 26 00 1c 01 01 01 00 20 00 0a 00 02 09 07 03 10
   05 88 26 53 00 0a 07 03 11 05 88 26 03 00 00
```

Refer to *Transfer Request for IAM - Message Breakdown (2-118)* for an explanation of this message.

2. The MTP3 transmits an IAM on the SS7 link.

3. The MTP3 on Stack 0 receives an IAM and sends a Transfer Indication (0x14) for IAM from component HMDT (0x002B) to the host.

```
X->H
00 44 00 43 00 25 04 00 01 08 01 00 00 2b 00 14 01 03 00
   2b 00 30 04 25 00 0c 00 00 00 03 00 00 00 06 05 00 02
   00 04 26 00 1c 01 01 01 00 20 00 0a 00 02 09 07 03 10
   05 88 26 53 00 0a 07 03 11 05 88 26 03 00 00
```

Refer to *Transfer Indication for IAM (2-119)* for an explanation of this message.

4. The host sends a Transfer Request (0x0A) for ACM to the matrix for Component MTP3 HMRT (0x002C) on stack 0.

```
H->X
00 2e 00 44 00 29 04 00 01 08 01 00 00 2c 00 0a 01 03 00
   2b 00 1a 04 25 00 0c 00 00 00 06 00 00 00 03 05 00 02
   00 04 26 00 06 01 01 06 10 00 00
```

5. MTP3 transmits ACM on the SS7 link.

6. MTP3 on Stack 1 receives an ACM and sends a transfer indication (0x14) for ACM from component HMDT (0x002B) to the host.

```
X->H
00 2e 00 43 00 26 04 00 01 08 01 01 00 2b 00 14 01 03 00
   2b   00 1a 04 25 00 0c 00 00 00 06 00 00 00 03 05 00 02
   00    04 26 00 06 01 01 06 10 00 00
```

7. The host sends a Transfer Request (0x0A) for ANM to the Matrix for Component MTP3 HMRT (0x002C) on stack 0.

```
H-X
```

```
00 2c 00 44 00 2a 04 00 01 08 01 00 00 2c 00 0a 01 03  00
   2b 00 18 04 25 00 0c 00 00 00 06 00 00 00 03 05 00  02
   00 04 26 00 04 01 01 09 00
```

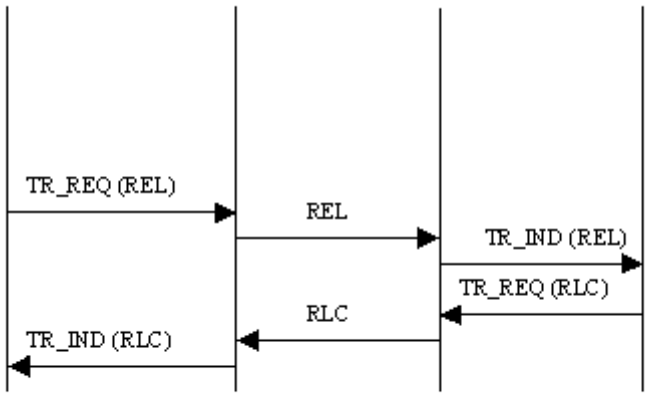
8. MTP3 transmits an ANM on the SS7 link.

9. MTP3 on Stack 1 receives an ANM and sends a Transfer Indication (0x14) for ANM from component HMDT (0x002B) to the host.

```
X-H
00 2c 00 43 00 27 04 00 01 08 01 01 00 2b 00 14 01 03 00
   2b  00 18 04 25 00 0c 00 00 00 06 00 00 00 03 05 00 02
   00  04 26 00 04 01 01 09 00
```

10. The call is established.

Call Release



1. Send transfer request (0x0a) for REL to matrix for Component MTP3 HMRT (0x2c) on stack 1.

```
H-X
00 2c 00 44 00 00 04 00 01 08 01 01 00 2c 00 0a 01 03 00
   2b 00 18 04 25 00 0c 00 00 00 03 00 00 00 06 05 00 02
   00 04 26 00 04 01 01 12 00
```

2. MTP3 on Stack 0 receives REL and sends a transfer indication (0x14) for REL from component HMDT (0x2b) to the host.

```
X-H
```

```
00 2c 00 43 00 1a 04 00 01 08 01 00 00 2b 00 14 01 03 00
   2b 00 18 04 25 00 0c 00 00 00 03 00 00 00 06 05 00 02
   00 04 26 00 04 01 01 12 00
```

3. Send transfer request (0x0a) for RLC to matrix for Component MTP3 HMRT (0x2c) on stack 0.

```
H-X
00 2c 00 44 00 00 04 00 01 08 01 00 00 2c 00 0a 01 03 00
   2b 00 18 04 25 00 0c 00 00 00 06 00 00 00 03 05 00 02
   00 04 26 00 04 01 01 10 00
```

4. MTP3 on Stack 1 receives RLC and sends a transfer indication (0x14) for RLC from component HMDT (0x002B) to the host.

```
X-H
00 2c 00 43 00 1b 04 00 01 08 01 01 00 2b 00 14 01 03 00
   2b 00 18 04 25 00 0c 00 00 00 06 00 00 00 03 05 00 02
   00  04 26 00 04 01 01 10 00
```

Service Indicator

The Service Indication part of the messages below differs depending on the protocol as follows.

- SNM - 0
- SNM_MAINT - 1
- SCCP - 3
- TUP - 4
- ISUP - 5
- DUP_CIRC - 6
- DUP_REG - 7

Transfer Request for IAM - Message Breakdown

```
00 44          'length
00 44          'message type
00 28          'seq
04             'node id
00 01 08 01 01 'stack aib
00 2c 00 0a    'mtp transfer requestS
01             'number of icbs
03             'icb type - extended
00 2b          'icb subtype - raw_msu_data
00 30          'length
04 25          'parameter name
00 0c          'length
00 00 00 03    'dpc
00 00 00 06    'opc
```

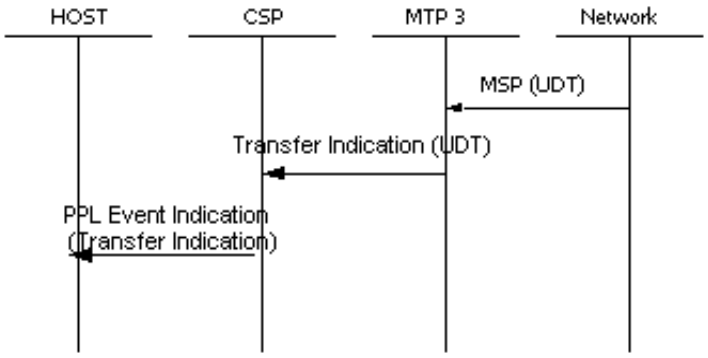
```
05              ‘service indicator
00              ‘message priority
02              ‘network indicator
00              ‘signaling link set
04 26           'parameter name
00 1c           'length
01 01           ‘CIC number 0x01 01
01              'msg id (IAM)
00              ‘MF parameter 1 - Nature of Connection
20 00           ‘MF parameter 2 - Fwd Call Indicator
0a              ‘MF parameter 3 - Calling Party’s Category
00              ‘MF parameter 4 - Trans Medium Requirement
02              ‘pointer to MV parameter
09              ‘pointer to optional parameter
07              ‘length of MV parameter
03 10 05 88 26 53 00
                ‘MV parameter - Called Party Number
0a              ‘opt. param. 1 name - Calling party number
07              ‘optional parameter 1 length
03 11 05 88 26 03 00
                ’optional parameter 1 value
00              ‘end of optional parameters.
```

Transfer Indication for IAM

```
00 44           'msg length
00 43           'msg type
00 25           'seq
04              'node id
00 01 08 01 00 'stack aib
00 2b 00 14     'transfer indication
01              'number of icbs
03              'icb type - extended
00 2b           'icb subtype - raw_msu_data
00 30           'length
04 25           'parameter name
00 0c           'length
00 00 00 03     'dpc
00 00 00 06     'opc
05              ‘service indicator
00              ‘message priority
02              ‘network indicator
00              ‘signaling link set
04 26           'parameter name
```

```
00 1c          'length
01 01          ‘CIC number Ø1 Ø1
01             'msg id
00             ‘MF parameter 1 - Nature of Connection
20 00          ‘MF parameter 2 - Fwd Call Indicator
0a             ‘MF parameter 3 - Calling Party’s Category
00             ‘MF parameter 4 - Trans Medium Requirement
02             ‘pointer to MV parameter
09             ‘pointer to optional parameter
07             ‘length of MV parameter
03 10 05 88 26 53 00
               ‘MV parameter - Called Party Number
0a             ‘optional parameter 1 name - Calling party
               number
07             ‘optional parameter 1 length
03 11 05 88 26 03 00
               ‘optional parameter 1 value
00             ‘end of optional parameters.
```

TCAP Call Flow The following call flow is for an incoming TCAP call.



API Message

```
begin:(send ppl req for UDT from stack1)
message:

(00 3b 00 44 00 00 00
00 01 08 01 01
00 2c 00 0a
01 03 00 2b 00 27
04 25 00 0c 00 00 00 01 00 00 00 02 03 00 02 00
04 26
00 13
09 80 03 07 0b 04 43 01 00 82 04 43 02 00 92 03 0a 0b 0c)
```

```
response:(00 07 00 44 00 00 00 00 10)
report:

(00 3b 00 43 00 00 00
00 01 08 01 00
00 2b 00 14
   01 03
   00 2b
   00 27
   04 25
      00 0c
      00 00 00 01
      00 00 00 02
      03 00 02 00
   04 26
      00 13
      09 80 03 07 0b 04 43 01 00 82 04 43 02 00 92
      03 0a 0b 0c)

report_wait:(4000)
onfail:(exit)
end:(send ppl req for UDT from stack1)

begin:(send TR.REQ for UDTS from stack0)
message:

(00 3b 00 44 00 00 00
00 01 08 01 00
00 2c
00 0a
01 03
00 2b
00 27
04 25
   00 0c
   00 00 00 02
   00 00 00 01
   03 00 02 00
04 26
00 13 0a 80 03 07 0b 04 43 01 00 82 04 43 02 00 92 03 0a
   0b 0c)

response:(00 07 00 44 00 00 00 00 10)
report:(00 3b 00 43 00 00 00 00    'tr. ind. for UDTS.
01 08 01 01
00 2b 00 14
01 03 00 2b 00 27
04 25 00 0c 00 00 00 02 00 00 00 01 03 00 02 00
04 26
```

```
00 13 0a 80 03 07 0b 04 43 01 00 82 04 43 02 00 92 03 0a
   0b 0c)
report_wait:(40000)
onfail:(exit)
end:(send TR.REQ for UDTS from stack0)
```

SS7 Raw Data to Host

Overview

The SS7 Raw Data to Host feature allows the customer a way of tracking all Message Signaling Units (MSUs) and Link Status Signaling Units (LSSUs) transmitted to the network and received from the network.

This feature allows the SS7 Series 3 and SS7 PQ cards to send PPL event indications containing LSSUs and MSUs (received from the network as well as sent to the network) to the host. The *PPL Configure* (0x00D7) message is used to enable and disable (default) the feature.

Configuring the Feature

This feature is initiated by setting (enabling) configuration bytes 0x10 in the following PPL components:

- Enabling the PPL event indications containing the raw LSSUs for transmission/reception to and from the network is done by enabling the configuration bytes 0x10 in the MTP2 LSC and MTP2 IAC PPL components.
- Enabling the PPL event indications containing the raw MSUs for transmission/reception to and from the network is done by enabling the configuration bytes 0x10 in MTP3 HMDT and HMRT PPL components.

When the feature is enabled all the supported LSSUs and MSUs are reported to the host in the PPL event indication.

LSSUs Supported

The supported LSSUs are as follows:

MPT2 IAC (0x0022)

- Status Indicator of Out of alignment (SIO)
- Status Indicator of Normal (SIN)
- Status Indicator of Emergency (SIE)

MPT2 LSC (0x0023)

- Status Indicator of Out of Service (SIOS)
- Status Indicator of Processor Outage (SIPO)

PPL Changes

This feature supports the following PPL changes:

- MTP3 HMDT (0x002B)
- MTP3 HMRT (0x002C)

- MTP2 IAC (0x0022)
- MTP2 LSC (0x0023)
- Atomic Function 98

3 SS7 over the EXNET® Ring

Purpose This chapter describes the software architecture and configuration necessary for the implementation of SS7 over the EXNET® ring, which enables SS7 control of circuits across nodes in a CSP. Dialogic has developed SS7 over the ring to provide a system-level integration of the industry-standard SS7 technology into the CSP. The CSP product and SS7 protocol together provide the capability for signaling links to be physically terminated on a single node, called the SS7 Server Node. Thus, SS7 over the ring allows control and handling of those links to be processed on a single or redundant pair of SS7 cards located within that node. The actual voice channels or Circuit Identification Codes (CICs) being controlled by the signaling links can originate or terminate on any node, local or remote, within the domain controlled by the SS7 Server Node in the CSP.

SS7 Over the Ring Introduction

Overview

The SS7 over the EXNET® ring software enables the use of four SS7 stacks on the ring. Only voice data travels on the ring to remote nodes. Communication between nodes is done over the Ethernet for all command and control information, which allows connection of the voice path on the ring, so that EXNET® is fully available for maximum voice traffic.

Important! When the term SS7 card appears in this text, it applies to both the SS7 PQ card and the SS7 Series 3 card.

Conditions

The following application conditions apply to an installation and operation of SS7 over the ring:

- An SS7 PQ card supports up to four nodes through a single Signaling Point (SP).
- An SS7 Series 3 card supports up to seven nodes through a single Signaling Point (SP).
- A maximum of four SS7 protocol stacks per SS7 card (or pair)
- Required termination of the signaling links on the SS7 Server Node

SS7 cards that are operating in a redundant card pair as a single SP must be installed in adjacent slots in the same chassis.

Single Server Node

Using modified system software, a single Server Node (up to four nodes on an SS7 PQ card, up to seven nodes on an SS7 Series 3 card) can appear to function in the network as a single Signaling Point (SP) with a single Originating Point Code (OPC).

Single Server Node implementation of SS7 over the ring offers the following advantages:

- An increased number of voice channels supported by a single SP with the redundancy offered by the EXNET® ring architecture.
- No changes to host API messages
- The same level of functionality provided with the SS7 Fault Tolerant Redundancy.

SS7 Over the EXNET® Ring Software

When a call is setup, the signaling data is carried using SS7 out-of-band from the voice channels. Dedicated Ethernet links are required at the SS7 connection in order to get the signal from the signaling network to

the channels, ensure synchronization, and provide instructions. In summary, the signaling data commands the phone to ring and gives instructions on the SS7 network. This information is shared with all other remote nodes using the Ethernet connection.

Features

The SS7 over the ring software is functionally equivalent to the existing SS7 card, including the protocol variants that are supported, the limitations, API messaging, configuration, and other functions.

- The SS7 over the ring software can be used to control CICs in remote nodes in a CSP.
- The SS7 software offers increased processing power.

Applications

CSP

Within Dialogic's CSP, a primary requirement for Central Office environments is enabled: a fully redundant architecture with optional redundancy at the card level.

Each node in a multiple node CSP supports up to 2048 non-blocking ports in:

- Central Office environments
- Distributed switching
- Wireless Local Loop (WLL)
- In-building wireless applications
- Service platforms
- Wireless infrastructure
- Long distance, debit card, callback, and call center operations

SS7

Within an SS7 network architecture there are three major components:

Service Switching Points (SSP) are CSPs that perform call processing on calls that originate, tandem, or terminate at the CSP.

Signal Transfer Points (STP) are CSPs that relay messages between network CSPs and databases.

Service Control Points (SCP) are CSPs that contain centralized network databases, which provide enhanced services.

The SS7 card permits the CSP to act as an SSP in an Advanced Intelligent Network (AIN). The SS7 card supports the following two architectures:

American National Standards Institute (ANSI)

International Telecommunications Union - Technology Sector (ITU-TS)

SS7 Over the Ring and Single Server Node Communications

Overview

The SS7 stack(s) resides on a single SS7 card in a non-redundant configuration or on a pair of SS7 cards in a redundant configuration. The card(s) is located in a single CSP node (the SS7 Server Node) that provides the common channel signaling for up to seven nodes in a CSP environment with SS7 Series 3 cards and up to four nodes with SS7 PQ cards.

- Multiple stacks on an SS7 card (or redundant pair) can exist in the SS7 Server Node. Each stack operates independently as a separate SP with its unique Originating Point Code (OPC).
- Voice circuits controlled by SS7 can originate or terminate on any of the seven nodes within the domain of a single SS7 Server Node. Voice traffic between the nodes in the CSP use the EXNET® ring.

Ethernet

The SS7 card (or pair) communicates with the CSP Matrix Series 3 Cards in remote nodes via Ethernet, which is used for:

- Call control among CSP nodes
- Host connectivity
- SS7 call control for SS7 over the ring

CCS I/O Cards

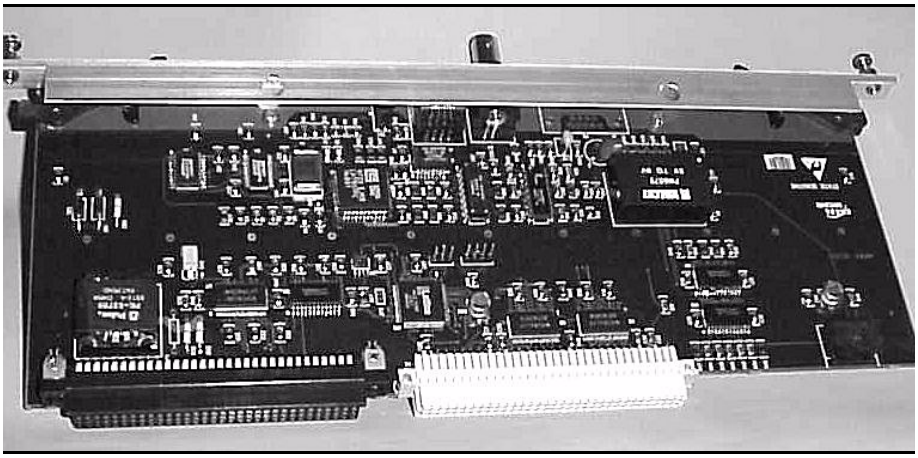
The I/O cards handle the additional communications for SS7 hardware in SS7 over the ring. These I/O cards are a redundant card pair occupying two card slots in the CSP on the node and using an HDLC link between them for communication. The SS7 card requires the CCS I/O card. The SS7 Series 3 card requires the SS7 Series 3 I/O card.

Features:

- The I/O cards are single slot.
- Each I/O card supports a single 10 Base T or 10 Base 2 Ethernet connection (according to an auto-select function, which detects the active Ethernet connection).
- Each I/O card is paired with an SS7 card.

The CCS I/O card shown in *Figure 3-1, CCS I/O card (3-6)*, is described fully in the CCS I/O HPD.

Figure 3-1 CCS I/O card



CCS I/0 Cards and HDLC

Separate HDLC buses are used in the following ways:

- Midplane HDLC bus for local communication (including matrix/SS7) within a node
- CCS I/O Redundancy Cable, an independent HDLC link, for redundancy between SS7 cards

SS7 Call Control at the matrix level (on the SS7 Server Node) communicates with the Matrix Controller on remote CSP nodes over the Ethernet via the CCS I/O card. The same physical network is used for host and matrix controller communications.

Redundancy

Redundant configurations are optional. If you choose to configure redundancy, removing one of a pair of CCS I/O cards in a Server Node will not interrupt SS7 system operation, nor will it require any user intervention.

Important! A redundant SS7 PQ card connects to the remote matrix controller cards through the Ethernet, and to the primary SS7 PQ card through the HDLC bus.

Card Failure

A CSP with SS7 capability can withstand the following failure conditions while maintaining stable calls:

- Matrix controller switchover in the SS7 Server Node: If a matrix controller card CSPs over in an SS7 Server Node, remote voice circuits are not affected, including calls that have been answered or are in the process of being answered.

No remote calls are purged and incoming calls can continue to be processed and set up to remote nodes, ensuring that a voice circuit is formed.

- Matrix controller switchover in a node other than the SS7 Server Node
- A single SS7 card failure/removal in a redundant SS7 configuration
- A single CCS I/O card failure in a redundant SS7 configuration
- CSP ring failure with alternate paths available

Important! As a result of EXNET® Ring failure, voice circuits either originating or terminating on nodes that are isolated from each SS7 Server Node will be affected.

SS7 over the EXNET® Ring

Overview

The signaling stack for the SS7 protocol interfaces with another network via SS7 signaling links on the SS7 card. Each stack on each SS7 card in an EXNET® Ring must have its own unique address, called a Point Code, assigned to the SS7 stack

Before SS7 Over the Ring

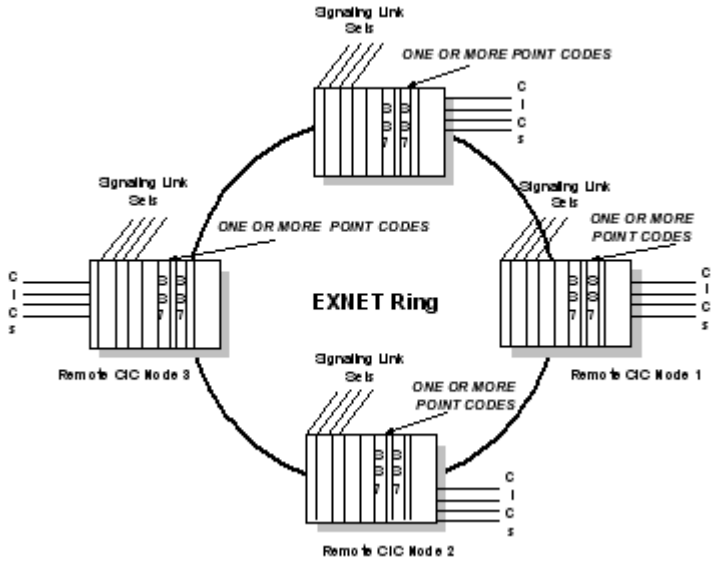
Using single nodes

Prior to SS7 over the EXNET® ring, if there were four remote nodes in the system, each node would have a Point Code assigned for the SS7 stack and there would be at least four Point Codes on the ring *Figure 3-2, Previous Methodology for Point Codes (Ethernet not shown) (3-9)*. To the network, each CSP appeared as a separate entity, and the CSP was not really functioning as a single SP. This reduced the efficiency in routing, particularly in defining address locations where connections have been made. This could have led to problems in data transfer and to time delays. Cost was also a major concern. Since each SP required separate signaling links, along with redundant links, the monthly lease costs became prohibitive.

Diagram

The next figure shows the Point Code assignments before implementation of SS7 over the ring.

Figure 3-2 Previous Methodology for Point Codes (Ethernet not shown)



Conditions **before** implementation of SS7 over the ring included:

- One or more SP/nodes
- One or more Link Set per node
- One or two SS7 cards per node

Using SS7 Over the Ring with a Single Server Node

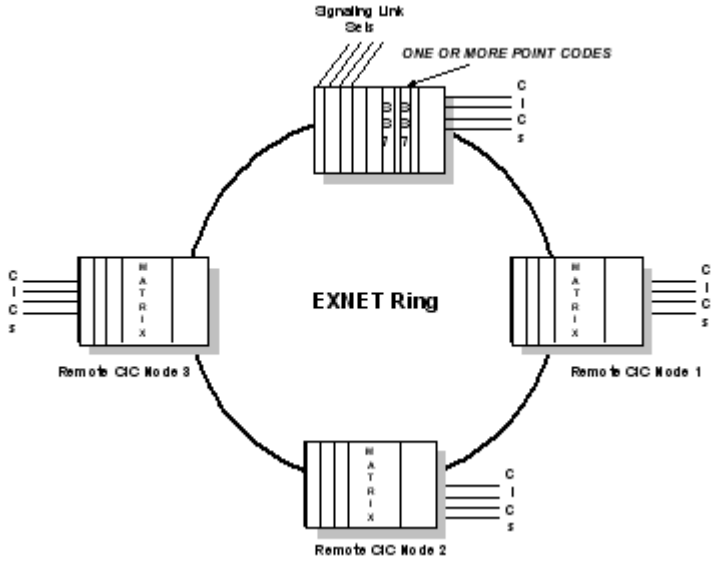
With SS7 over the ring, using only the Server Node for SS7 connections to the network, **only one Point Code is required per Server Node.** To the network, the CSP at the Server Node finally appears as a single SP.

In the new SS7 over the ring implementation, the capability exists to get a single link set to that external point. Instead of handling each signaling link differently, now all the links can be combined as one signaling link set, improving routing implementation and reducing monthly link lease costs.

Diagram

The next figure shows the Point Code assignments after implementation of SS7 over the ring:

Figure 3-3 SS7 Over the Ring Methodology for Point Codes in a Single Server Node (Ethernet not shown)



Conditions after implementation of SS7 over the ring on an SS7 card include:

- One per 4-node CSP
- Four per 7-node CSP
- One or more Link Set(s) per 4-node CSP; a minimum of Four Link Sets per 7-node CSP
- One or 2 card pair(s) per four-node CSP; a minimum of four card pairs per 7-node CSP

Using SS7 Over the Ring with Multiple Server Nodes

In a CSP system there can be multiple Server Nodes on the same EXNET® ring, each with its associated remote nodes (CICs), for a maximum total of seven nodes. Each Server Node operates the same way as it does in the Single Server Node configuration. Control is limited to a maximum of four nodes comprised of one Server Node and three remote nodes (CICs).

This means that each Server Node appears as a separate or mini-network to the PSTN, and each Server Node continues to have its own unique Stack IDs, Point Codes, and remote nodes (CICs). A multiple node setup on the EXNET® therefore contains four different Point Codes.

This feature does allow calls to be connected between two CICs that reside on different SS7 Server Node mini-networks, since all the Server Nodes are part of the same CSP system.

Conditions and Usability

Redundancy is not affected.

A fault or matrix controller switchover on one Server Node in a multiple node system will not affect any node that is controlled by another Server Node in the same system.

Limitations

- Each stack on each Server Node in a multiple node system must have a unique Point Code. No two stacks in the entire CSP system may share the same Point Code.
- Each remote (CIC) node must be controlled by one and ONLY one SS7 Server Node. Two Server Nodes should not be sharing control over CICs on the same remote node.

CAUTION

Dialogic strongly recommends that two Server Nodes not share control over CICs on the same remote node. It will cause unexpected results.

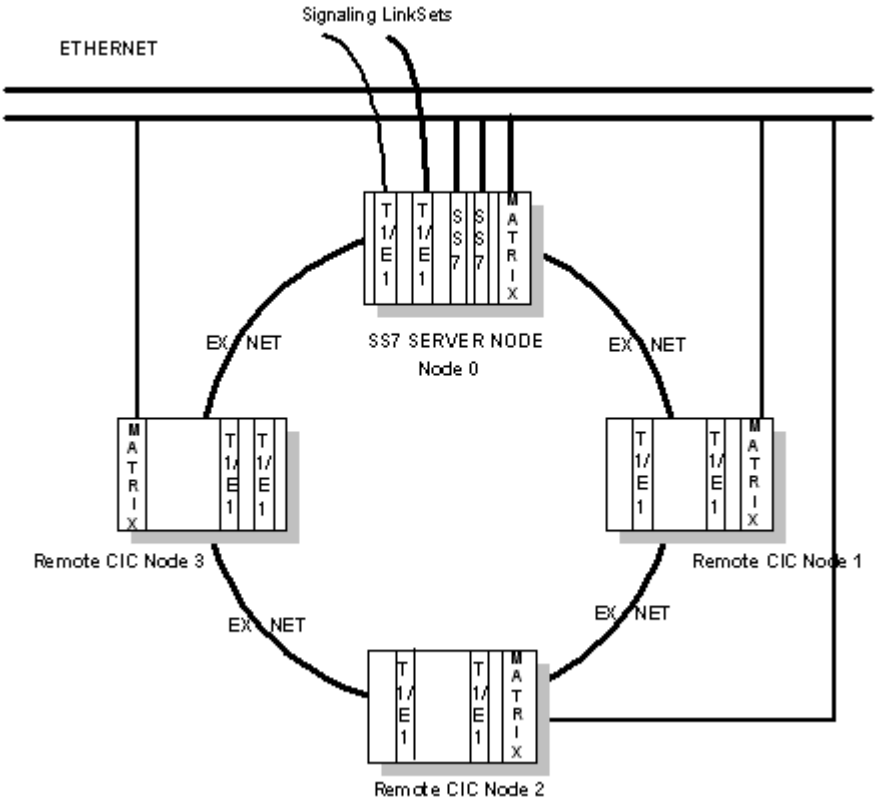
Single Server Node Configuration

One to four stacks can be configured for each SS7 card or SS7 card pair that has been installed on the SS7 Server Node. This means there can be from one to four SPs configured, each with its own OPC. (Configuration of a 7-node ring is done the same way at each of the four Server Nodes.)

Diagram

The next figure shows an example of a Single SS7 Server Node configuration:

Figure 3-4 SS7 Over the Ring with Ethernet and Exnet® Connections



In the previous figure one node has been selected as the Server Node and used to command the SS7 resources and distribute and share them with the three remote nodes. No SS7 cards need to be installed into the CSPs on the remote nodes. Only the Server Node has the SS7 card or redundant pair. Voice channels (CICs) in all four CSP nodes are controlled from the Server Node.

Selection of the Server Node

Considerations that may affect the decision include:

- High availability
- Available capacity and accessibility for connecting signaling links
- Available capacity, particularly the number of slots, for SS7 cards and card pairs

Signaling Links

All signaling links **must** terminate on the Server Node. In previous topologies with CSP, SS7 signaling links had to communicate with an SS7 card on every node to manage control of voice circuits.

L3P on the SS7 card(s) communicates with CSP Call Control on the active matrix controller cards of both local and remote nodes for call processing, using Ethernet communication paths (see previous figure).

Configuring SS7 Over the Ring

Purpose

This section describes the procedure for bringing SS7 over the ring into service. It assumes that all hardware configuration and cabling is complete, that all nodes are powered up, and that a host-to-node communication link has been established.

The *CSP Developer's Guide: Overview* provides detailed setup procedures for downloading system software, for establishing socket connections and for configuring rings and nodes.

Before you begin

With SS7 over the ring, the *CCS Redundancy Configure* (0x5B) message will take at least 50 seconds before the ACK is returned, assuming there is no call processing.

To ensure that SS7 over the ring is operational before configuration starts, do the following:

1 Modify the application, as required, to support SS7 control of CICs in multiple chassis/nodes that have a single Originating Point Code (OPC) in the Server Node.

2 Move signaling links to the SS7 Server Node. Change the configuration as required.

3 Insert a new, RISC-based SS7 card(s) into the SS7 Server Node

4 Install the CCS I/O cards, with the CCS I/O Redundancy Cable (see CCS I/O Redundancy Cable Installation and Removal" in the *Hardware Installation and Maintenance Manual*).

5 Execute the Configuration Procedure for SS7 over the ring.

Guidelines

When you configure SS7 over the ring, perform the steps below, and use these steps as a guideline to your initial configuration.

All SS7 configuration messages must be sent to the node that contains the SS7 card(s).

Configuration Procedure

Follow the procedure below to configure SS7 over the ring.

1 Assign the logical node ID.

Send the *Assign Logical Node ID* message (0x10), using the serial number of the CSP chassis number as the physical node ID.

2 Assign logical spans throughout the CSP.

Send the *Assign Logical Span ID* message (0xA8) to assign spans to physical span offsets from EXNET®.

3 Configure the spans and channels corresponding to SS7 signaling links for clear channel.

Send either the *E1 Span Configure* message (0xD8) or the *T1 Span Configure* message (0xA9), depending on the protocol you use, to establish clear channel signaling.

4 Assign the SS7 cards as either primary or secondary (optional).

Send the *CCS Redundancy Configure* message (0x5B) if you are using a redundant card pair.

No CSP-specific configuration is required.

5 Configure the SS7 Signaling Stacks with a stack ID, an Originating Point Code (OPC), and the variant to be used.

Send the *SS7 Signaling Stack Configure* message (0x5C) to assign the stack ID, OPC, and variant.

6 Configure the signaling link sets with a signaling link set ID to a given adjacent point code.

Send the *SS7 Signaling Link Set Configure* message (0x5D) to configure the link sets.

No CSP-specific configuration is required.

7 Configure the signaling links with the logical span/channel.

Send the *SS7 Signaling Link Configure* message (0x5E) to configure the links. The timeslot is provided off the local bus.

8 Configure the signaling routes and specify the relationship between a remote Destination Point Code (DPC) and the signaling links capable of reaching DPC.

Send the *SS7 Signaling Route Configure* message (0x5F) to configure the routes.

No CSP-specific configuration is required.

9 Assign the voice circuits and associate a set of span/channels with a set of Circuit Identification Codes (CICs) for a particular stack ID.

Send the *SS7 CIC Configure* message (0x6A) to configure CICs.

Based on the logical span ID, the node that owns the logical span is informed that the stack ID controlling the voice circuits is the stack assigned to control the CICs in the message.

10 Send *Service State Configure* for each of the following:

- To bring: the spans into service.
- To bring the SS7 links into service.
- To bring the channels (CICs) into service.

END OF STEPS

Communication between SS7 stacks and CSP Call Control components

The remote nodes know which stacks are controlling their local voice channels by a *SS7 Signaling Stack Configure* message received from the SS7 card.

With this information, CSP Call Control can communicate with the stack controlling the voice channels, and the SS7 stacks can communicate with any remote CSP Call Control components, based on the fact that every node knows the global span assignments, which determine local and remote connections. To make connections in the CSP ring, all nodes must have information about the spans on all other nodes. Since each span is given a **unique** ID, and all call connections

occur via the combined information of channel and span IDs, global span assignment information is essential to every node. See the *Connect* (0x00) message in the *API Reference*.

ISUP Remote Control

Overview

The ISDN User Part (ISUP) Remote Control feature is essentially the same as SS7 over EXS, but without the Exnet® ring. An SS7 server node contains the SS7 cards that control the channels (for example, CICs) in the local and remote nodes. ISUP Remote Control differs from SS7 over EXS as follows:

- There is no Exnet® ring
- CICs must originate and terminate in the same CSP node.

The ISUP Remote Control feature provides the software architecture and configuration necessary an SS7 card or card pairs in one CSP node to control the ISUP CICs in up to seven CSP nodes.

ISUP defines the protocol and procedures used to set-up, manage, and release trunk circuits that carry voice and data calls over the Public Switched Telephone Network (PSTN).

For example, when the CSP server node is provisioned, the host provisions the remote CSP nodes that communicate with it. This allows multiple CSPs to act as a logical switch in order to share resource information. A CSP, for example, with the SS7 component configured, can assign Circuit Identification Codes (CICs) to any span/channel of a remote CSP.

The actual voice channels or CICs, being controlled by the signaling links, must originate and terminate on the same node within the domain controlled by the Server Node.

Providing SS7 signaling within an CSP system essentially means that a local or remote SS7 stack can control voice circuits located within any node. This is accomplished by providing Ethernet connectivity between the host and remote CSPs in order to support messaging between L4 and the TDM to provide bearer traffic.

A CSP system with an SS7 stack is viewed by the network as a single Origination Point Code (OPC). The OPC is the address of the SS7 stack (network).

SS7 Card Requirement

For use with the following cards:

- SS7 Series 3 card

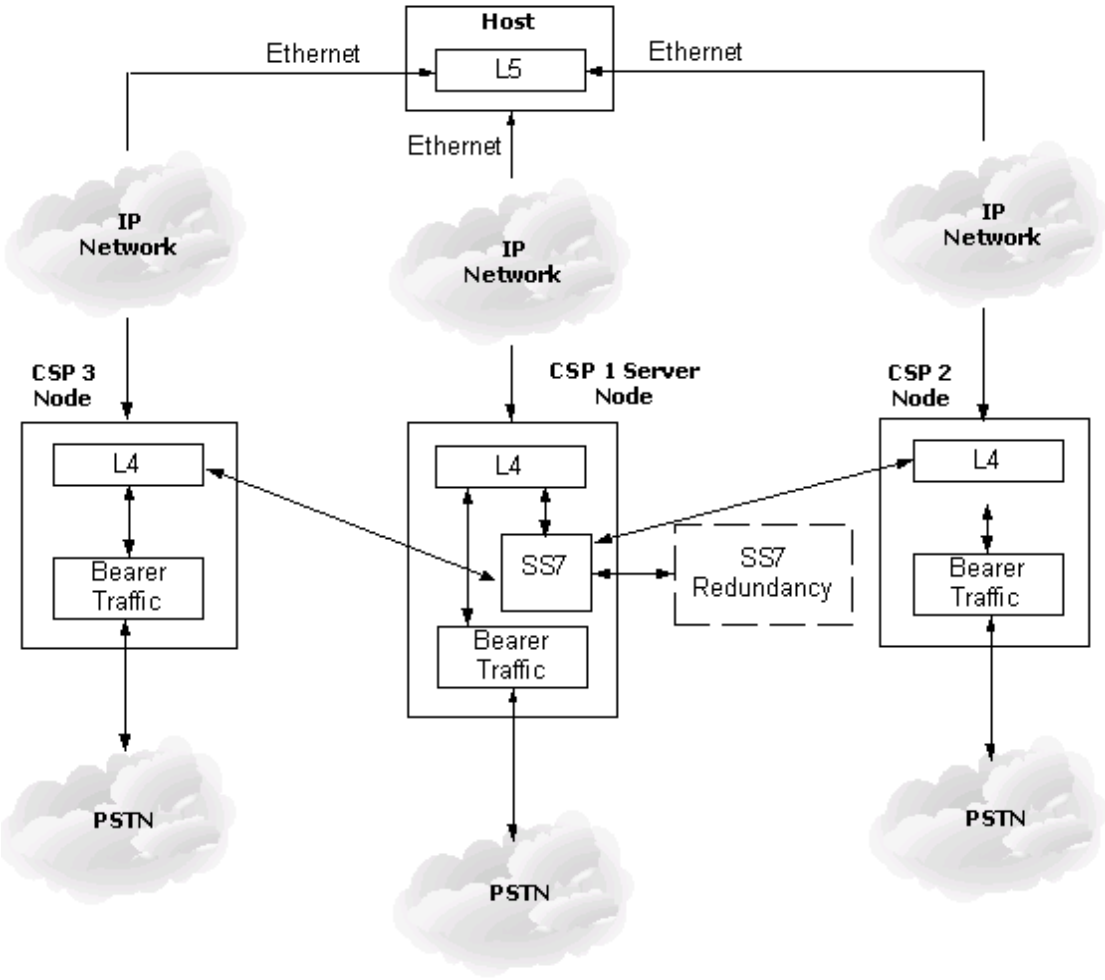
- CCS I/O Series 3 card

Maximum Node Requirement

A maximum of seven nodes is supported.

Host to Multi-Node Configuration

The following diagram illustrates a host controlled multi-node CSP system. Located in the Server Node, the SS7 stack is capable of controlling local as well as remote voice circuits. The SS7 signaling links terminate in the Server Node and the actual voice circuits can terminate within any node.The Ethernet connectivity enables the SS7 stack to layer L4 on remote nodes where the CICs reside.



Important! In a multinode system, if the SS7 card(s) loses ethernet connectivity, it takes approximately 50 seconds for the SS7 card(s) to send out BLO or CGB to the carrier(s) for all the CIC residing in the other EXNETed nodes.

Configuring the SS7 Stack

Purpose

This section describes the procedure for bringing the SS7 stack into service. It assumes that all hardware configuration and cabling is complete, that all nodes are powered up, and that a host-to-node communication link has been established.

The *CSP Developer's Guide: Overview* provides detailed setup procedures for downloading system software, for establishing socket connections and for configuring nodes.

Before you begin

To ensure that the SS7 stack is operational before beginning the configuration procedure, do the following:

1 Modify the application, as required, to support SS7 control of CICs in multiple nodes that have a single Originating Point Code (OPC) in the Server Node.

2 Move the SS7 signaling links to the Server Node. Change the configuration as required.

3 When you configure the SS7 stack, perform the steps below, and use these steps as a guideline to your initial configuration.

Configuration Procedure

Follow the procedure below to configure the SS7 stack.

1 Assign the logical node IDs to CSPs.
Send the *Assign Logical Node* message (0x0010), using the serial number of the CSP chassis number as the physical node ID.

2 Set the Enable ISUP Remote Control (0x01) bit on all nodes (server and remotes) of a multi-node system using the *Assign Logical Node* message (0x0010).

3 Download System Software license. Send *Product License Download* (0x0079) message.

4 Assign logical spans throughout the CSP.

Send the *Assign Logical Span ID* message (0x00A8) to assign spans to physical span offsets.

5 Download SS7 Software license. Send *Product License Download* (0x0079) message.

6 Configure the spans and channels corresponding to SS7 signaling links for clear channel.

Send either the *E1 Span Configure* message (0x00D8) or the *T1 Span Configure* message (0x00A9), depending on the protocol you use, to establish clear channel signaling.

7 Configure the SS7 Signaling Stacks with a stack ID, an Originating Point Code (OPC), and the variant to be used.

Send the *SS7 Signaling Stack Configure* message (0x005C) to assign the stack ID, OPC, and variant.

8 Configure the signaling link sets with a signaling link set ID to a given adjacent point code.

Send the *SS7 Signaling Link Set Configure* message (0x005D) to configure the link sets.

9 Configure the signaling links with the logical span/channel.

Send the *SS7 Signaling Link Configure* message (0x005E) to configure the links. The timeslot is provided off the local bus.

10 Configure the signaling routes and specify the relationship between a remote Destination Point Code (DPC) and the signaling links capable of reaching DPC.

Send the *SS7 Signaling Route Configure* message (0x005F) to configure the routes.

11 Send the *SS7 CIC Configure* message (0x006A) to the Server Node. Assign the voice circuits and associate a set of span/channels with a set of Circuit Identification Codes (CICs) for a particular stack ID.

Send the *SS7 CIC Configure* message (0x006A) to configure CICs.

Based on the logical span ID, the node that owns the logical span is informed that the stack ID controlling the voice circuits is the stack assigned to control the CICs in the message.

12 Send *Service State Configure* for each of the following:

- To bring: the spans into service.
- To bring the SS7 links into service.
- To bring the channels (CICs) into service.

Frequently Asked Questions

Overview

The following questions and answers address common issues related to SS7 over the ring.

Questions

How does the 50-second delay at start-up affect operation of SS7 Over the Ring?

The 50-second delay is a one-time delay during initialization only. It occurs when the host sends a *CCS Redundancy Configure* (0x5B) message and waits for the ACK, which takes 50 seconds. See also the *CCS Redundancy Configure* message in the *API Reference*, which adds this note: "Dialogic highly recommends that, when you configure SS7 redundancy, you send this message first, followed by all other SS7 configuration messages."

How is synchronization affected if configuration is attempted before the delay is complete?

The host cannot and should not attempt to perform any other SS7 configuration until synchronization is finished. This is no different from the current implementation of SS7 redundancy.

Are there any limitations for PPL-related actions on the CSP? What happens, for example, when the PPL Event Request has to address voice circuits where they reside, on the remote node?

For all PPL-related API messages that use an AIB addressing the logical span and channel of an SS7 CIC, the Logical Node ID field MUST indicate the node ID of the node on which the CIC resides.

The PPL API messages affected are as follows:

- *PPL Event Request* 0x44
- *PPL Timer Configure* 0xCF
- *PPL Configure* 0xD7
- *PPL Assign* 0xD1
- *PPL Data Query* 0xDE
- *PPL Audit Query* 0xDD

The API messages mentioned previously are affected when they are paired with one of the following SS7 CIC-based components:

SS7 Components		
0x0F L3P_CIC	0x11 L3P_TUP	0x16 ISUP_BLS
0x14 ISUP_CCI	0x15 ISUP_CRI	0x19 ISUP_CRR
0x17 ISUP_BLR	0x18 ISUP_CRS	0x1C ISUP_CGRR
0x1A ISUP_UCIC	0x1B ISUP_CGRS	0x1F ISUP_HGBR
0x1D ISUP_GBUS	0x1E ISUP_GBUR	0x44 ISUP_CRCR
0x42 ISUP_HLB	0x43 ISUP_HRB	0x47 ISUP_HGBS
0x45 ISUP_MGBS	0x46 ISUP_MGBR	0x78 ISUP_CVS
0x76 ISUP_CQS	0x77 ISUP_CQR	0x81 ISUP_CRCS
0x79 ISUP_CVR	0x80 ISUP_CCO	0x52 TUP_CPC
0x82 ISUP_DCO	0x83 ISUP_CRO	0x56 TUP_CCI
0x54 TUP_BLR	0x55 TUP_BLS	0x59 TUP_CGRR
0x57 TUP_CRI	0x58 TUP_CRS	0x5C TUP_MBUR
0x5A TUP_CGRS	0x5B TUP_MBUS	0x5F TUP_SBUS
0x5D TUP_HBUS	0x5E TUP_HBUR	
0x60 TUP_SBUR	0x12 ISUP_CPC	

For any other combination of PPL-related API messages and SS7 PPL components, the Logical Node ID field **must** be set to the node ID of the SS7 server node.

In general, if an SS7 CIC is being addressed, the Logical Node ID field should indicate the location of that CIC. If any other object is being addressed (for example, slot, SS7 link, route), the Logical Node ID field should indicate the SS7 Server Node.

All PPL Event Indications from one of the above components will contain the Logical Node ID of the node on which the CIC resides.

How does redundancy operate in SS7 Over the Ring?

From a host software standpoint, SS7 over the ring redundancy operates in exactly the same manner as the current implementation of SS7 redundancy.

Does SS7 Over the Ring require an IP address on the card?

No. With SS7 over the ring, the SS7 card communicates using a Dialogic proprietary protocol that is similar to a UDP and which does not use an IP address. The host does not need to communicate with the

card directly for any reason. The Ethernet connectivity on SS7 over the ring enables the SS7 card to communicate with the Matrix Controller cards on remote nodes where the CICs reside.

When CSP receives the undefined Protocol message from adjacent CSP, what does CSP do?

CSP will reply UPU message, send PPL indication “UPU MSU transmitted” (PPL 0x2C, ID 0x06) and discard the message.

When CSP receives the undefined ISUP Messages from adjacent CSP, what does CSP do?

CSP will reply with a CFN message with Cause 97 (message type non-existent or not implemented), send PPL indication “Undefined message received” (ppl 0x13, id 0x03), and discard the message.

When CSP receives the undefined Parameters in ISUP Message form adjacent CSP, what does CSP do?

CSP will reply CFN message with cause 99 (information element non-existent or not implemented), and send PPL indication “parameter discard” (ppl 0x12, id 0x4e) and continue the message without this undefined parameter included.

What is the maximum number of CIC groups that can be created per node?

128

FYI: ITU defines MCP and PCP for undefined messages and parameters. MCP and PCP messages provide support for unrecognized signaling as defined by ISUP 1992 (ITU-T White Book). This support becomes necessary, for example, if the exchange receives unrecognized signaling information from a later version of ISUP. With MCP/PCP, the exchange can take the appropriate action and ensure its compatibility with new signaling messages, no matter which version they may be.

4 SS7 Call Control for ISUP

Purpose This chapter describes Dialogic support of Integrated Services Digital Network User Part (ISUP). The chapter includes information on several of the commonly used ISUP messages and provides details about ISUP segmentation and outgoing and incoming setup.

More Information See the Appendix at the end of this book for tables of ISUP-supported messages, Field IDs for supported fields and restrictions on the implementation of ISUP messages.

ISUP Introduction

Overview

The Integrated Services Digital Network User Part (ISUP) is used for establishment of wired connections between exchanges. It includes messages associated with the connection and disconnection of calls. ISUP is the protocol used to support the signaling necessary to provide voice and non-voice services in telephone communications. It is an extension of SS7, used as the interface protocol for voice and data within, and for ingression or egression to/from the Public Switched Telephone Network (PSTN.)

Advantages of ISUP

There are numerous advantages to using ISUP for call processing and trunk group maintenance/management functions. These advantages include the following:

- Faster call setup
- Conservation of network resources
- Improved customer feature support as well as enhanced features
- Improved/Automated trunk management procedures

ISUP requires that the exchange be digital and support SS7. This does not mean that the exchange cannot interface with non-SS7 capable devices or exchanges. All that is required is that the exchange be able to interwork with other signaling types such as TUP, R2, R1, DTMF and/or SS7.

Definition

The ISDN User Part defines the protocol which is used to support the signaling functions required for both a) Non-ISDN voice/data communications and b) ISDN voice/data communications in North America, and for ISDN functionalities throughout the rest of the world. ISUP can also be used to support dedicated/private telecommunications systems consisting of either digital or analog or mixed networks. The data from ISDN messages is transferred to ISUP messages for call processing or data transfer. They are two separate protocols that function together. ISDN can be thought of as a user-side protocol while ISUP is strictly network side. ISDN may also be used between exchanges, but that is not the primary intent of the protocol. There is enough flexibility and spare capacity within the protocol to support present application requirements as well as, any foreseeable future requirements.

Currently the ISDN User Part is designed to use the services of the SS7 Message Transfer Part (MTP). The use of the SS7 Signaling Communication Control Part Network has been provided for in the specification but, at present, no procedures exist that require the use of the SCCP for ISUP. ISUP connects to the SS7 MTP as User Part 5. Because ISUP exists as a User Part, all Layer 4, 5, 6, and 7 functions are handled within the software that comprises ISUP.

End User Signaling

The exchange is required to provide connectivity and service to the end users of the service by the exchange. This end user's ISDN connectivity is usually provided by the use of Primary Rate Interface. While ISUP is not the protocol used to the end user, it does support the services and functionalities provided to the end user by ISDN and other end user protocols.

Primary Rate Interface

The Primary Rate Interface (PRI) is implemented differently depending upon whether the user is following ITU-T or ANSI standards. Under ITU-T standards, the PRI consists of:

- 30 (64Kb) channels for voice/data,
- One, 64Kb channel for signaling/data (channel 16)
- One 64Kb channel for framing/synchronization (channel 0).

Under ANSI standards, the PRI consists of:

- 23 channels (64 Kb) for voice/data
- one (64 Kb) channel for signaling/data (channel 9).

PRIs are typically used between an exchange and a PBX or other device requiring high data transfer rates. The PRI in most of North America is 23B+D (1.544 Mb) and is equal to DS1, and is normally carried on a T1 carrier. In the rest of the world, the PRI is 30B+D (2.048 Mb) and is carried on an E1 carrier. Other configurations are possible. In North America, a single D channel may support its 23 B channels plus 24 B channels from each of an additional three PRIs. The PRI offers subscribers flexible bandwidth, high rates of data transfer, and a very stable platform.

Inter-Exchange Signaling

ISUP is used for the exchange of data between exchanges. Data is passed in messages between exchanges for call processing, maintenance, and circuit management. These messages are in the standard SS7 format of the MSU. The actual ISUP data is carried within the SIF.

An understanding of these messages is important as it allows you to know what type of data is being transferred between exchanges at any given moment in time. There are many messages defined in the ITU and ANSI standards. However, very few exchanges, if any, support all of them. It is a good idea to be familiar with the various messages even if they are not supported by your exchange. It is possible that one of these unsupported messages may be received at your exchange, and it is important to know how to deal with it. See the ISUP messages and parameters.

ISUP Call Control and Circuit Management API

The host manages ISUP call control and circuit management through the following mechanisms:

- Common Call Control API messages, using SS7 Parameter (ISUP) Jibs to pass data.
- Channel State

  DS0 Status Change messages notify the host of service state changes and channel purges. The *Service State Configure* message allows the host to bring channels in and out-of-service.
- The *PPL Event Indication* and *PPL Event Request* messages allow the host to send and receive messages to and from PPL components.

The CSP sends *PPL Event Indication* messages to the host in response to various SS7-related call processing events, such as reception of an ACM, ANM, or CON (for ITU only) for ISUP.

Data is included in the SS7 Parameter ICB (ISUP). If you do not require this information, acknowledgement (ACK) the message and then ignore it.

The host can generate SS7-related call processing or management events using the *PPL Event Request* message.

You can modify the default call model to have the Called and Calling Party Numbers passed to the host as BCD encoded digits. See *SCCP/ TCAP (6-1)* for more information.

ISUP Incoming Call Setup

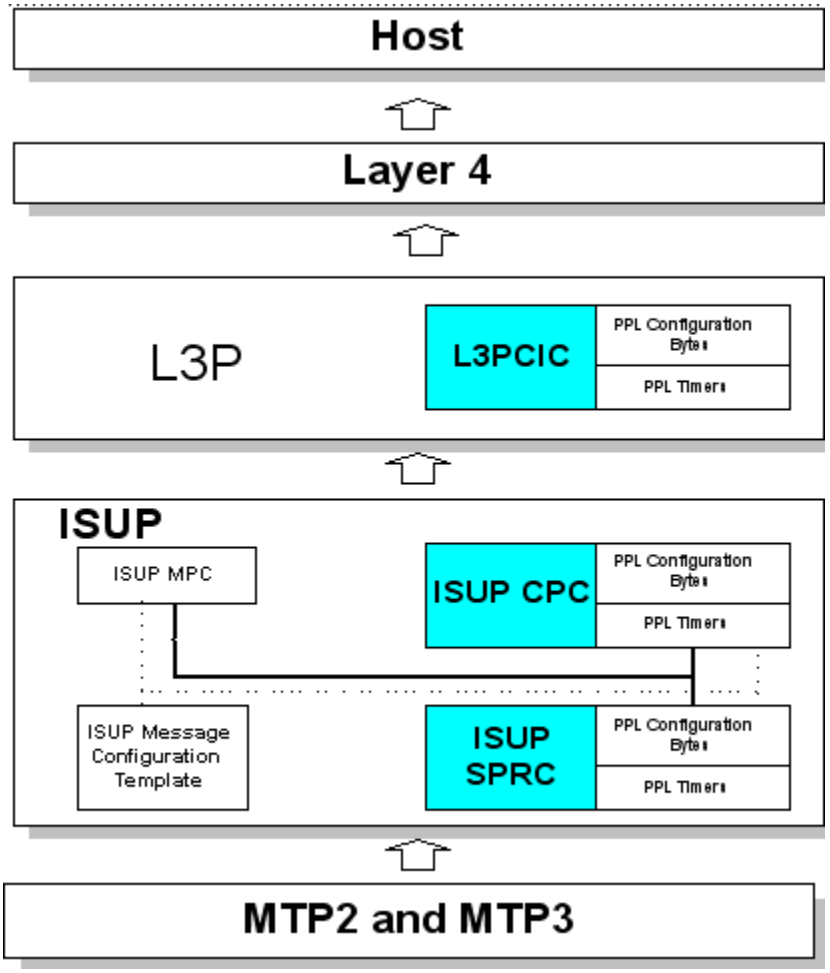
Overview

This section includes call flow examples for ISUP incoming call control.

Diagram of Incoming SS7 Call Handling

The figure below illustrates the default interaction between the host and the SS7 PPL components during an incoming SS7 call. Shading identifies the software components involved in the call.

Figure 4-1 Interaction Between Host and SS7 PPL Component



Sequence of Default Incoming Call

The default incoming call sequence shown in the diagram above is described in the table below.

Stage	Description
1	Raw ISUP messages are received by MTP and passed to ISUP.

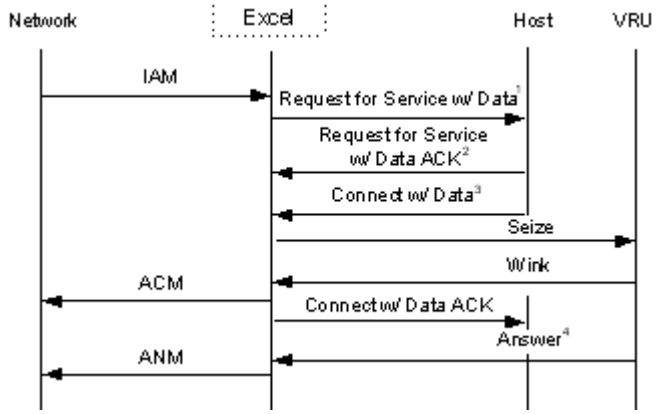
Stage	Description
2	ISUP SPRC is responsible for routing the data to the appropriate component. Assuming the data is for a valid CIC, it is routed to ISUP CPC for processing. If the data is maintenance related, it is routed to MPC. In the event of an unrecognized message type or invalid CIC, a CFN or UCIC is sent to the network (for ANSI only).
3	Assuming no error conditions occur, ISUP consults the ISUP Message Configuration Template to identify the message and translate the raw data into the format of an SS7 Parameters ICB to pass to L3P. To modify the ISUP message formats, the host issues the ISUP Message Configuration message. The host can configure which parameters are included and can also customize messages for ISUP variants.
4	L3P passes generic setup indication messages to Layer 4, which drive Layer 4 into an inseized state and generate a Request For Service With Data message to include an SS7 parameters ICB with all received parameters to the host. You can modify this format to specify the parameters passed or to have the Called and Calling Party passed to the host as BCD encoded digits.

Call Flows

The call flows in this section include more details about the numbered messages. The numbered messages are described following each call flow.

Basic Incoming Call Setup

The following call flow uses a Voice Response Unit (VRU) as the outgoing leg of a tandem connection. The VRU is a T1 E&M wink start channel. More details about the numbered messages are provided following the call flow.



1. When using the default L3P, all SS7 parameters in the IAM pass to the host in the *Request For Service With Data* message in the following format (see *Information Control Blocks* chapter in the *API Reference* for information on ICB formats):
   - Parameter Name
   - Parameter Length
   - Parameter Value

You can also configure L3P to send the called and calling party digits as BCD-encoded digit strings by modifying the PPL Configuration Bytes. (See *Request for Service* message format.)

2. The host must acknowledge the *Request For Service With Data* message or the CSP resends the message once after 5 seconds.
3. The host must provide mandatory SS7 parameters for the ACM and ANM as determined by the following:

   Connect With Data message with SS7 Parameters ICBs (the parameters for each must be sent in separate ICBs). For ANM and ACM, this condition allows the host to dynamically overwrite any pre-stored parameters in the L3P CIC Configuration Bytes.

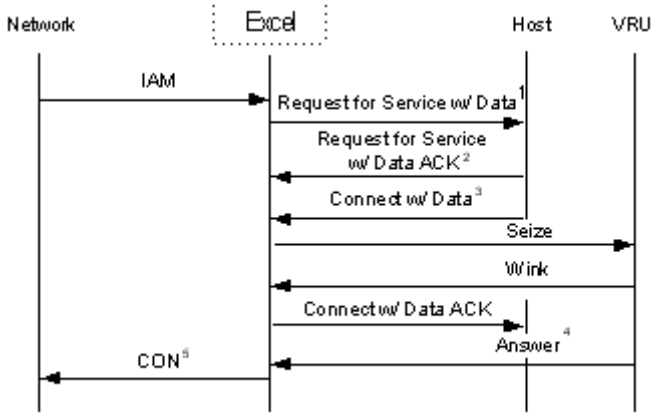
   Connect message using preprogrammed parameter values in the L3P CIC PPL Configuration Bytes.

   Generate Call Processing Event of Answer using pre-programmed ACM and ANM parameters.

   Propagate answer from VRU using pre-programmed parameters.
4. Because the Voice Response Unit (VRU) is running asynchronously to the CSP, Answer can return anytime after the Wink.

Incoming Call Setup Using CON (Instead of ACM/ANM - for ITU Only)

The following call flow shows a VRU as the outgoing leg of a tandem connection using CON. The VRU is a T1 E&M wink start channel.

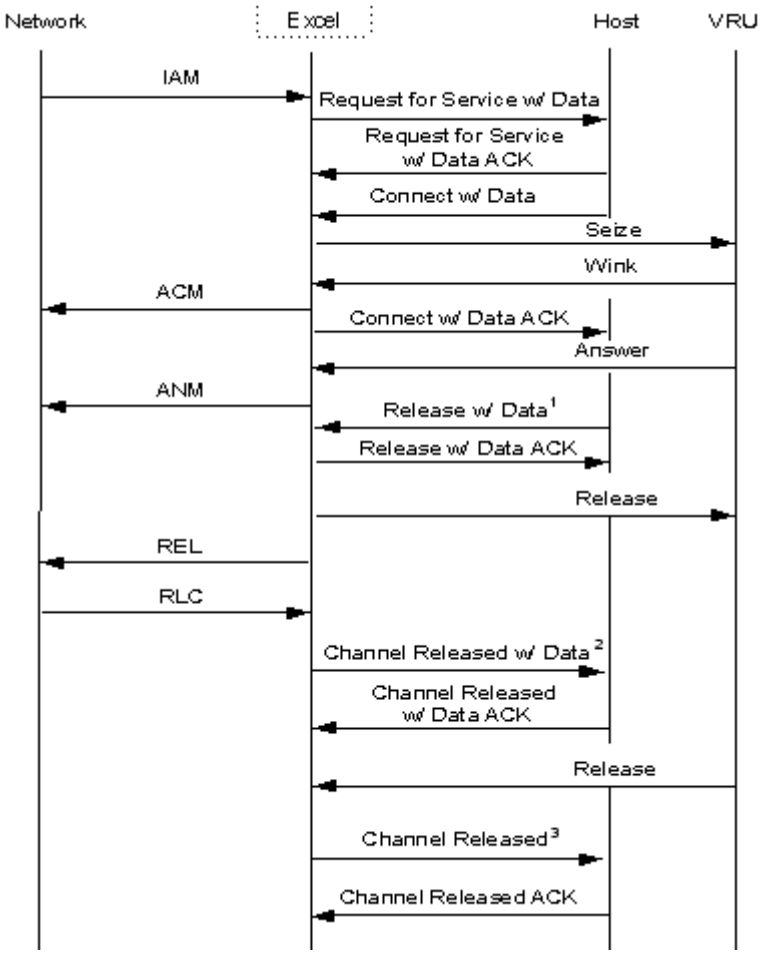


1. When using the default L3P, all SS7 parameters included in the IAM pass to the host in the *Request For Service With Data* message in the following format (see the *Information Control Blocks* chapter in the *API Reference* for information on ICB formats):
   - Parameter Name
   - Parameter Length
   - Parameter Value

You can configure the L3P to send the called and calling party digits as BCD-encoded digit strings by modifying the PPL Configuration Bytes (see *Request for Service Message* format.)

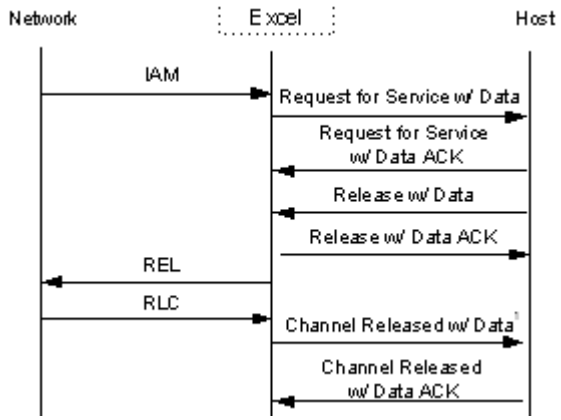
2. The host must ACK the *Request For Service With Data* message or the CSP will resend the message once after 5 seconds.
3. The host must provide the mandatory SS7 parameters for CON as determined by the following:
   - *Connect With Data* message with SS7 Parameters ICBs (the parameters for each must be sent in separate ICBs).
   - *Connect* message using preprogrammed parameter values in the L3P CIC PPL Configuration Bytes.
   - Generate *Call Processing Event* of Answer using defaults stored in CON parameter.
   - Propagate Answer event from the VRU using pre-programmed CON parameters.
4. Because the VRU is running asynchronously to the CSP, Answer returns anytime after the Wink.
5. Connect is set based on the specified setting in L3P CIC Configuration Byte 9.

Incoming Call Setup with Host-initiated Release



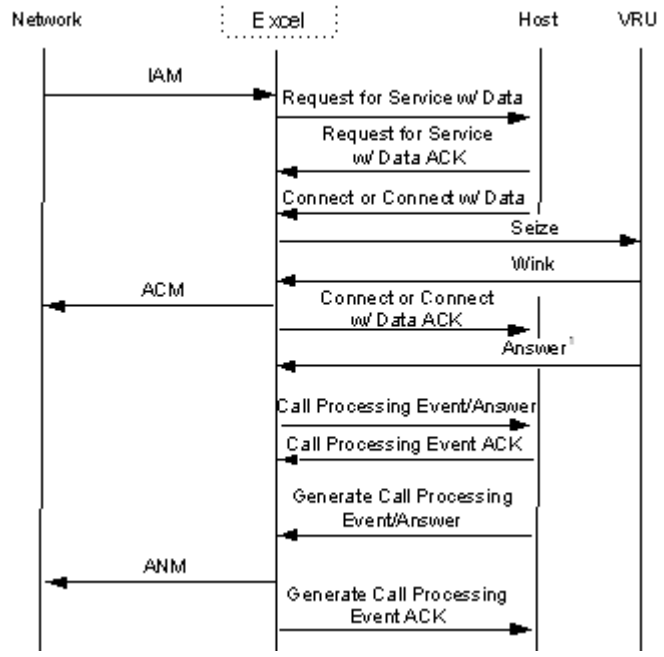
1. The host must provide at least the mandatory parameters for the forward REL, either of the following:
   - *Release With Data* message
   - *Release* message using preprogrammed parameter values in the L3P CIC PPL Configuration bytes
2. The *Channel Released With Data* message contains an SS7 parameter ICB with the parameters from the incoming RLC (Release Complete).
3. A *Channel Released* message indicates that the VRU channel is idle and ready for a new call.

Incoming Call Rejection by Host



1. The *Channel Release With Data* message contains an SS7 Parameter ICB with the REL parameters.

Incoming Call with Answer Supervision Mode of "Notify Only"



1. With Answer Supervision set to Notify Only, answer is not propagated to the SS7 channel. A *Call Processing Event* of Answer is sent to the host for the VRU.
2. To answer an SS7 call, the host may use the *Generate Call Processing Event* of Answer. The parameters for the ANM will be retrieved from the PPL Configuration Bytes.

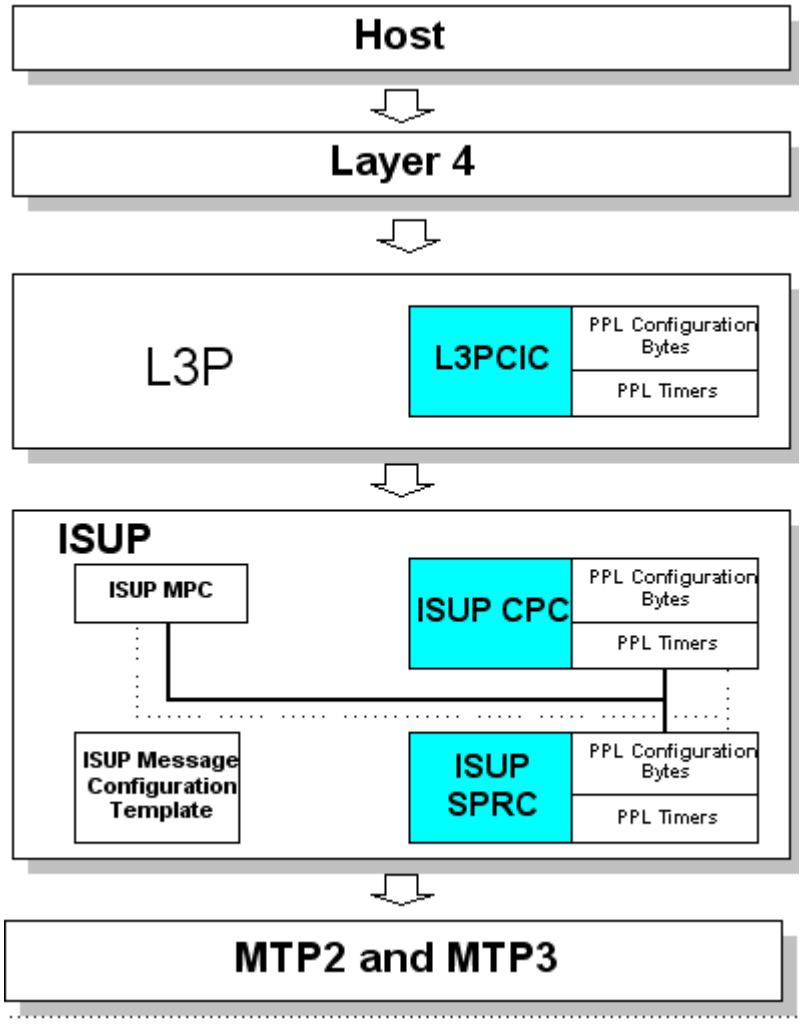
ISUP Outgoing Call Setup

Overview

This section includes call flow examples for ISUP outgoing call control.

Outgoing SS7 Call Setup Diagram

This diagram illustrates the interaction between the host and the SS7 PPL components in the handling of an outgoing SS7 call. Shading identifies the software components involved in the call.



Default Call Sequence

The table below describes the default outgoing call sequence.

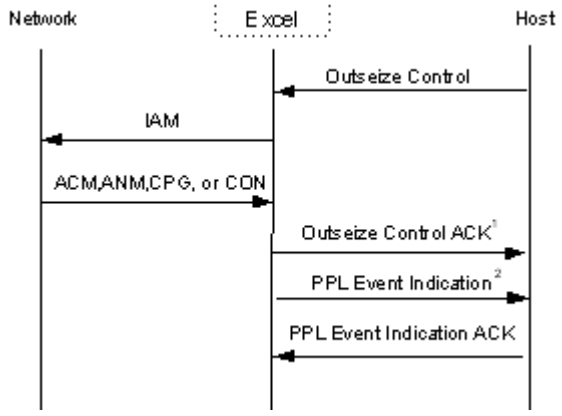
Stage	Description
1	An outgoing call is initiated by the host to the Outseize Control message. The CSP will accept address data as either BCD-encoded digits or in an SS7 Parameters ICB. The Outseize Control message drives L4 into an outseize state and an outseize request is sent to L3P. NOTE: The *Outseize Instruction List Configure* message is not supported for SS7. All outseize instructions must be included in the Outseize Control message.
2	L3P CIC translates the L4/L5 event into an ISUP event. Any parameters sent from L4/L5 are concatenated with pre-stored parameter data in the PPL Configuration Bytes.
3	The parameters and events are sent to the appropriate ISUP Component. If parameter validation is successful, the parameters are formatted into a raw SS7 message and sent to ISUP SPRC. If validation fails, a PPL Event Indication message of Protocol Violation is sent to the host.
4	ISUP SPRC routes the message to MTP for transmission.

Call Flows

The call flows in this section include more details about the numbered messages. The numbered messages are described following each call flow.

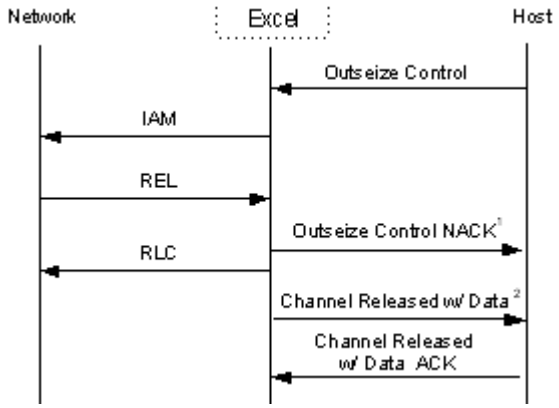
Basic Outgoing Call Setup

Any mandatory SS7 parameters not supplied by the host in an *Outseize Control* message are retrieved from the L3P CIC PPL Configuration Bytes.



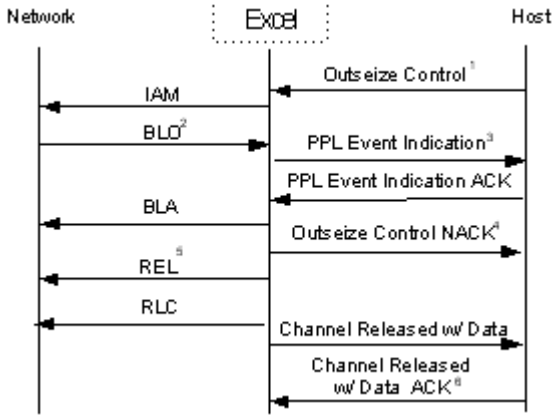
1. An *Outseize Control* message ACK does not report to the host until a backward signal is received from the network.
2. The *PPL Event Indication* message to the host indicates the backward signal type (ACM, ANM, CPG, or CON) and an SS7 Parameters ICB with parameter data. If the host application does not require parameter data, the CSP acknowledges and ignores the *PPL Event Indication* message.

Outgoing Call Rejection by Network



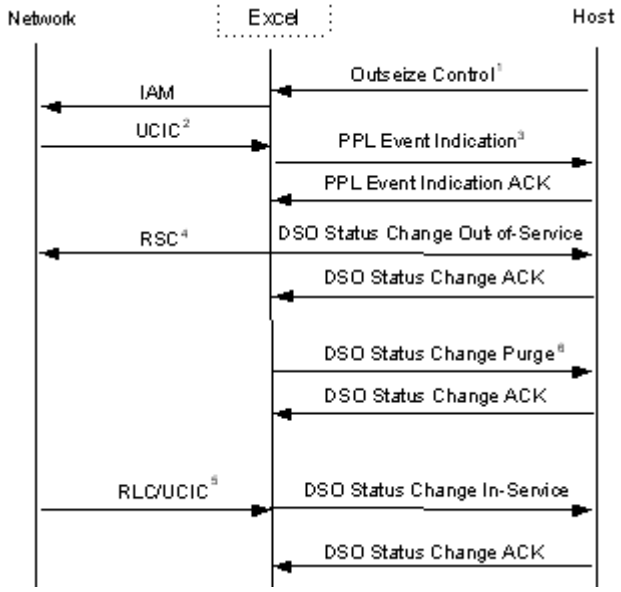
1. The host receives a Negative Acknowledgment (NACK) to the *Outseize Control* message.
2. The *Channel Released With Data* message contains an SS7 Parameters ICB with the REL parameter data.

BLO Received in Response to IAM



1. The *Outseize Control* message generates an outgoing IAM to the network.
2. The CSP receives BLO in response to IAM.
3. The CSP generates a *PPL Event Indication* message to the host, indicating receipt of BLO. The CSP automatically generates BLA to the network.
4. The CSP generates an *Outseize Control* message NACK to the host.
5. The CSP automatically generates REL to the network.
6. Upon receipt of the RLC from the network, the CSP generates *Channel Released with Data* message to the host.

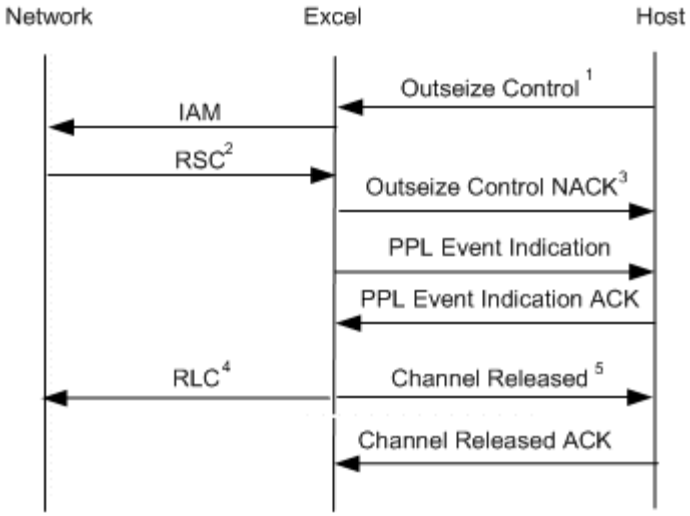
UCIC Received in Response to IAM (ANSI only)



1. *Outseize Control* message generates outgoing IAM to the network.
2. UCIC is received in response to the IAM.
3. *PPL Event Indication* is generated to the host indicating UCIC.
4. RSC is generated and purge is initiated by the CSP.
5. Upon receipt of RLC or UCIC from the network, DS0 status of In-Service is reported to the host.

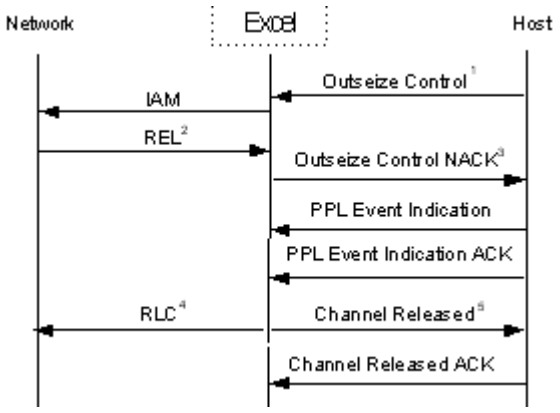
*Generation of RSC in this call flow is the default behavior. However, PPL Configuration Byte (0x2A) in L3P CIC can suppress this message. The RLC/UCIC response to the RSC would not be received. The CSP-to-host messages will be the same regardless of the configuration byte value.

RSC Received in Response to IAM



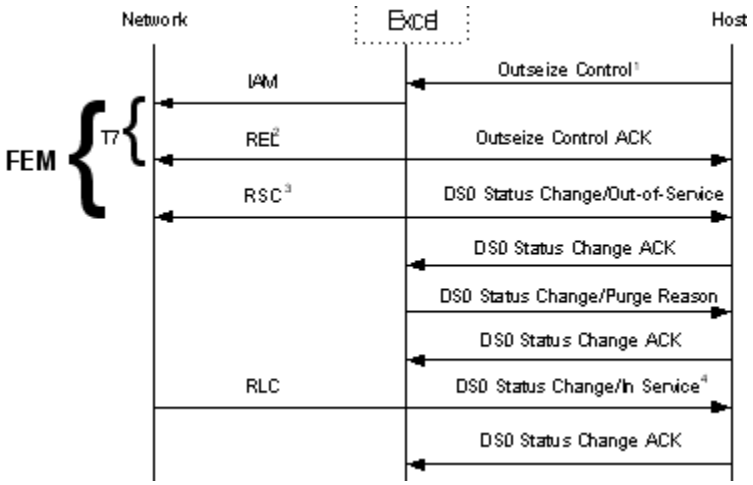
1. *Outseize Control* message generates IAM to the network.
2. RSC is received in response to IAM.
3. *Outseize Control* message ACK is generated to the host with response status of Outseize Failure No Acknowledgment (0x1B).
4. RLC is generated automatically to the network.
5. A *Channel Released with Data* message is sent, notifying the host that the channel is now idle.

No Acknowledgment of IAM (Expiration of T7)



1. The *Outseize Control* message generates an IAM to the network.
2. T7 expires. REL is generated automatically to the network.
3. An *Outseize Control* ACK is reported to the host with response status 0x1B (Outseize Failure No Acknowledge).
4. Upon receipt of RLC, a *Channel Released with Data* message is reported to the host.

No Acknowledgment of IAM or REL (Expiration of T7)

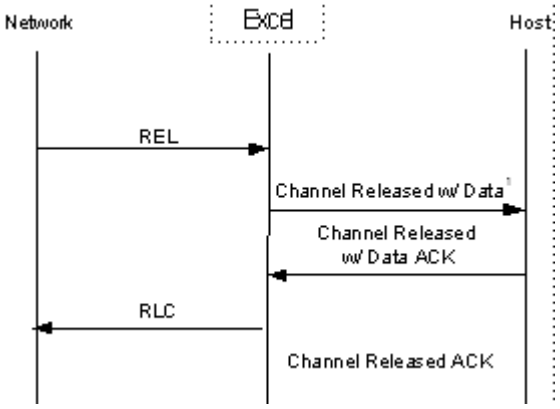


1. The *Outseize Control* message generates outgoing IAM to the network. Front End Machine FEM Outseize ACK Timer is initiated (29 seconds). T7 is initiated upon sending of the IAM.
2. T7 expires. REL is automatically generated to the network and *Outseize Control* ACK is reported to the host with a response of 0x1B (Outseize Failure No Acknowledgment).
3. FEM Outseize ACK Timer expires and initiates a channel purge. An RSC is generated to the network.
4. Upon receipt of the RLC, A *DS0 Status Change* indicating In-Service is reported to the host.

Network-Initiated Release

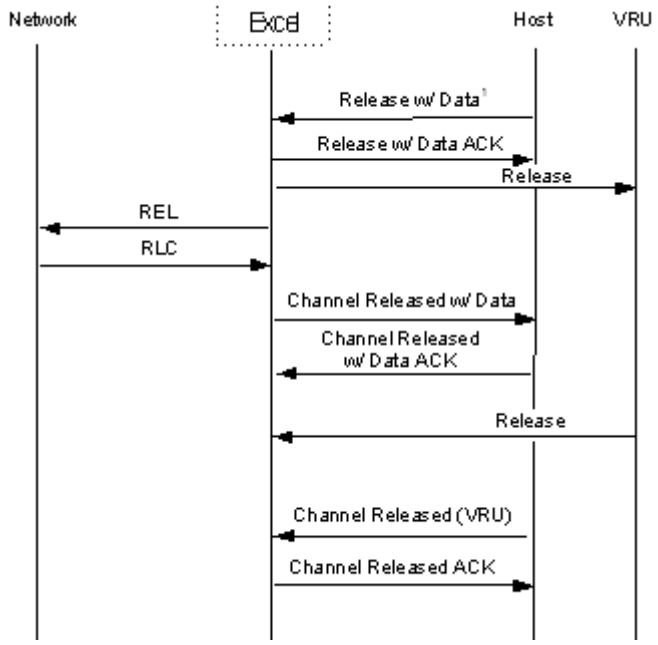
An SS7 network-initiated REL may occur at any time during call setup. The call flow is the same regardless of when the REL is received.

.



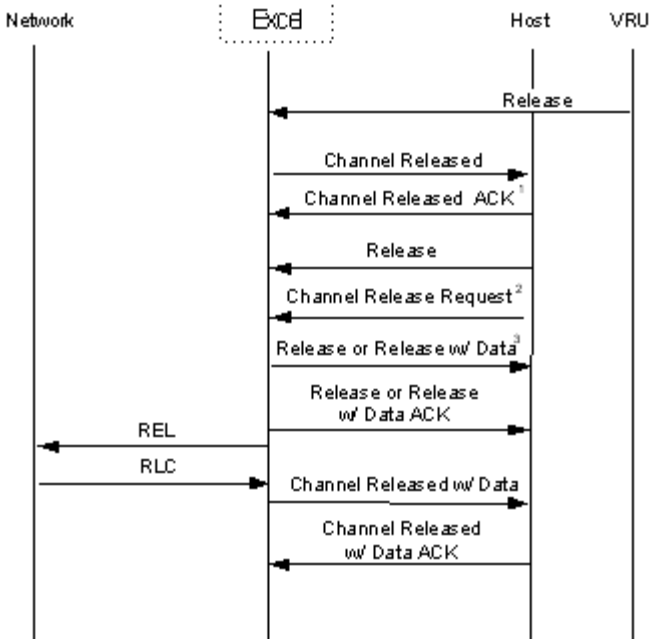
1. Parameters in the REL are sent to the host in the *Channel Released With Data* message.
2. The *Channel Released With Data* message must be acknowledged by the host, after which the channel is in an idle state and can be used for further call processing.

Host-Initiated Release



1. The host may provide parameters for the REL message in the *Release With Data* message.
2. The *Channel Released with Data* message must be acknowledged by the host within 5 seconds or the CSP resends the message.

VRU-Initiated Release



1. The *Channel Released* message must be acknowledged by the host within 5 seconds or the CSP resends the message.
2. If the Distant End Release Mode for the VRU and the Local End Release Mode of the SS7 channel is set to Release, the SS7 channel is not automatically released. The CSP sends a *Channel Release Request*, indicating that it intends to release the channel. If either mode is set to park, the SS7 channel parks and a *DS0 Status Change* message informs the host.

CIC Start-up Procedure

Overview

The CSP supports the Exchange Type A start-up procedure as defined by Q.767, Section 4.4.1. Type A does not send any messages at start-up, but responds to a GRS with a GRA, or an RSC with an RLC. As the default exchange type for the CSP is Type B, you must perform configuration steps to enable Type A.

Configuring Exchange Type

The exchange type is configurable with PPL Config Byte 60 of the SS7 ISUP SPRC component (0x13).

If the exchange type is set to Type B (0x01, the default), GRS/RSC is sent at start-up. If the Exchange Type is set to Type A (0x00), GRS/RSC is not sent at start-up.

Type B Procedure (Default)

The default exchange type for the CSP is Type B, which requires sending a GRS/RSC at start-up. The CSP waits for a remote GRA/RLC before bringing the CICs in service.

Type A Procedure

If the exchange type is set to Type A, GRS/RSC is not sent at start-up. Timer 4 of the L3P CIC component is initiated to wait for a remote GRS/RSC indication. The default timer value is 500 ms (5 s).

If the timer expires before GRS/RSC is received, the CICs are placed in service and a PPL event of 0x17 (No GRS/RSC Received, CICs In Service) is sent to the host. You can disable the sending of the PPL event to the host by changing PPL Config Byte 14 of the SS7 L3P CIC component (0x0F) to 0x00. (0x00 - Do Not Send PPL Event to Host; 0x01 - Send PPL Event to Host).

ISUP Segmentation

Overview

ISUP is associated with SS7-specific voice and data call processing. It is used to set up circuit connections and to maintain those connections between end-point subscribers for the duration of a call, using SS7-controlled Circuit Identification Codes (CICs). Segmentation is used to enable faster and more efficient transfer of ISUP messages across a network using the Segmentation Message (SGM). This feature is associated with the ITU White Book, 1993 (Q.761 - Q.764)

Definition

The ISDN User Part (ISUP) Segmentation provides the capability of breaking down larger packets of data into smaller ones in order to achieve compatibility with any network protocol that requires the smaller packet size. This process is necessary whenever large blocks of data need to be transmitted across a network, to offset problems with both time delays and error correction that could lead to traffic congestion. ISUP Segmentation in this way conserves critical network resources.

Using the Segmentation Message (SGM)

The Message Transfer Part (MTP) functions as the lowest layer in the SS7 protocol stack, and it is this layer that communicates with the rest of the network. But MTP cannot send more than 272 bytes in a single message. With ISUP there is a need to transfer larger amounts of data, so the SGM message was introduced to segment any ISUP message over 272 bytes into two parts. This process allows the MTP to transfer it successfully. The two parts include the primary ISUP message and the SGM, which contains the segmented portion over 272 bytes. The SGM is sent immediately after the primary ISUP message to the MTP layer.

Each message can only be segmented once. If the information in the message is so large that even an SGM becomes insufficient, then that information will be discarded.

At the receiving end, the primary ISUP message and the Segmented portion are concatenated before processing.

PPL Information

Simple Segmentation Control (SSC) Component ID (0x85)

This is the ITU ISUP PPL component created to support the SGM message in the CSP. This component takes care of the segmentation process at the transmission end and the re-assembly of the ISUP message at the receiving end.

CPC Component ID (0x12)

The ISUP CPC PPL component communicates through the SGM to the PPL SSC component.

ISUP Messages Used with SGM

The following messages are the only ones that can be sent or received using the SGM: IAM, ACM, ANM, CPG, and CON.

All other ISUP messages are transparent and are therefore passed through the SSC component without modifications or segmentation.

Size and Capacity

The host-to-CSP and CSP-to-host TCP message packet size is limited to 260 bytes and 492 bytes, respectively. Therefore, the ISUP message sent in a *PPL Event Request* from the host to the CSP cannot exceed 260 bytes.

When an ISUP message and its associated SGM are received by the PPL ITU SSC component, it will be divided into packets of 220 bytes each and then sent on to the L3P CIC PPL component.

Configuration

The outgoing SGM can be enabled or disabled based on a configuration byte setting. The CPC Configuration Byte for SGM control is “0x04”.

Configuration Bytes 0x01-0x05 are used for ISUP SCLC Component ID (0x85).

See *SS7 PPL Information* in this manual for more information on these configuration bytes.

Protocol Timers

Each PPL component has multi-purpose timers, which you can activate at any time. Each component using timers has a table that contains information on specific timers (name, value). When a protocol requires a timer, an atomic function is initiated to activate one of the PPL timers and to point to an index in the component’s timer table, which contains the value for the required timer.

One timer exists for the SSC PPL Component: 0x01

PPL Events

The PPL Indication “0x32” exists for the L3P CIC PPL Component (0x0F), for sending a Segmentation indication to the host.

Atomic Functions

SSC contains the atomic functions (51-63) that manage the segmentation and re-assembly of messages.

CPC contains the atomic functions (134-138) that communicate with the SSC component.

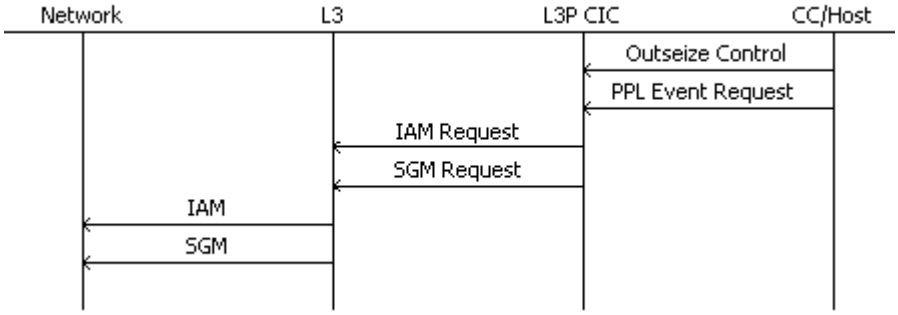
Redundancy

Redundancy for ISUP Segmentation is available.

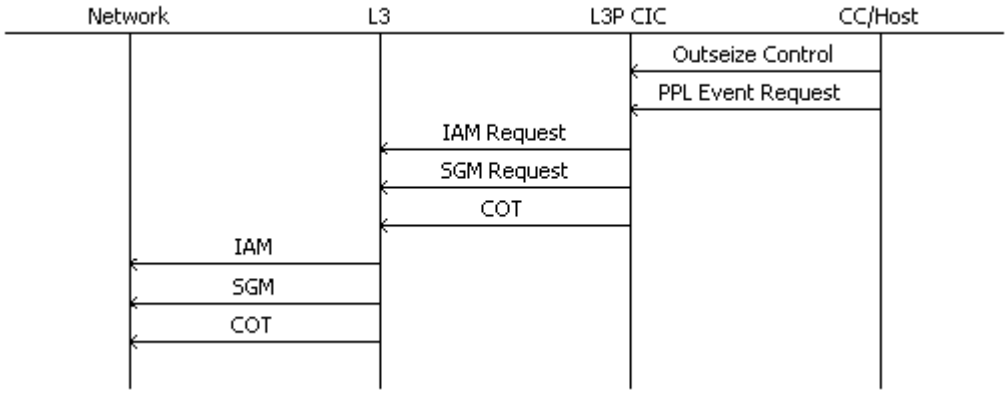
ISUP Segmentation Call Flows

Outgoing Call with Segmentation in IAM

The following call flow describes an outgoing call with Segmentation in IAM. After the host sends an Outseize Control message to L3P CIC, it then sends a PPL Event Request (0x60) to this component. After a setup request has been sent to CPC, and an IAM has been received by the network, an SGM message is sent to the network.

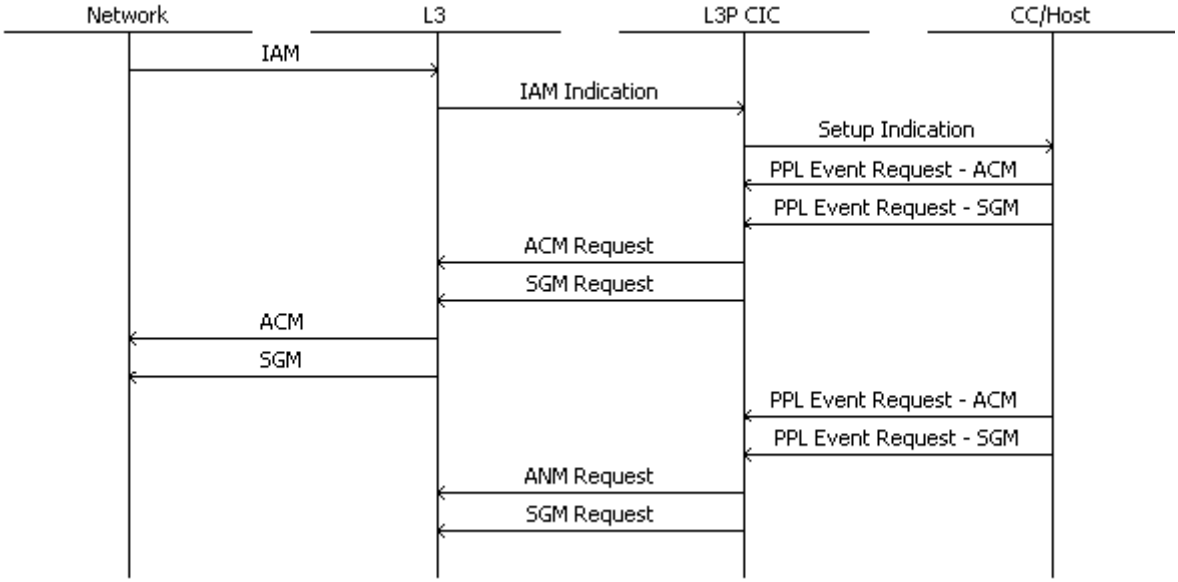


Outgoing Call with Segmentation in IAM when COT is enabled



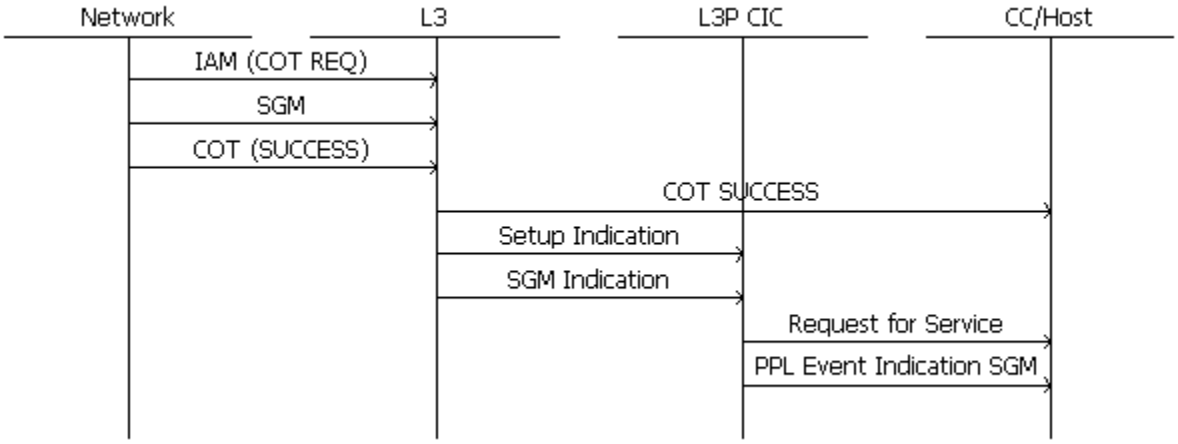
Segmentation in ACM/ANM

The following call flow describes segmentation in an ACM/ANM message.



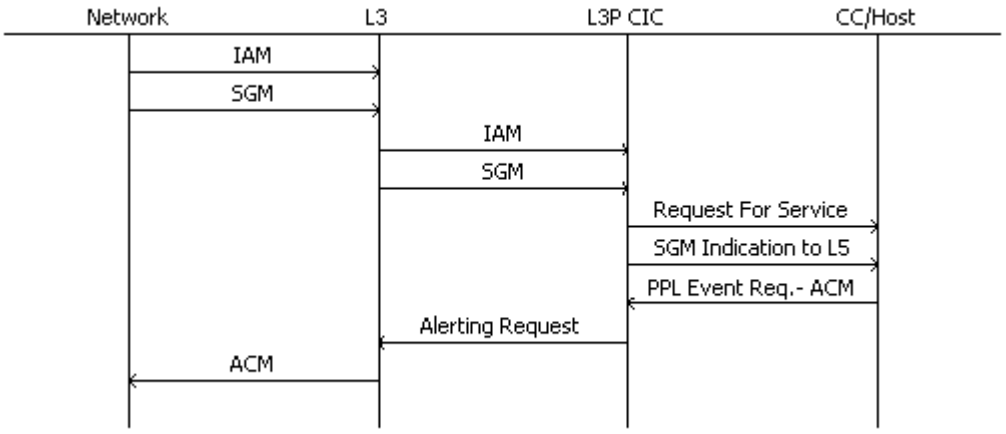
Incoming Segmentation with COT Enabled - COT Success

The following call flow describes an incoming call with segmentation and Continuity Check enabled.



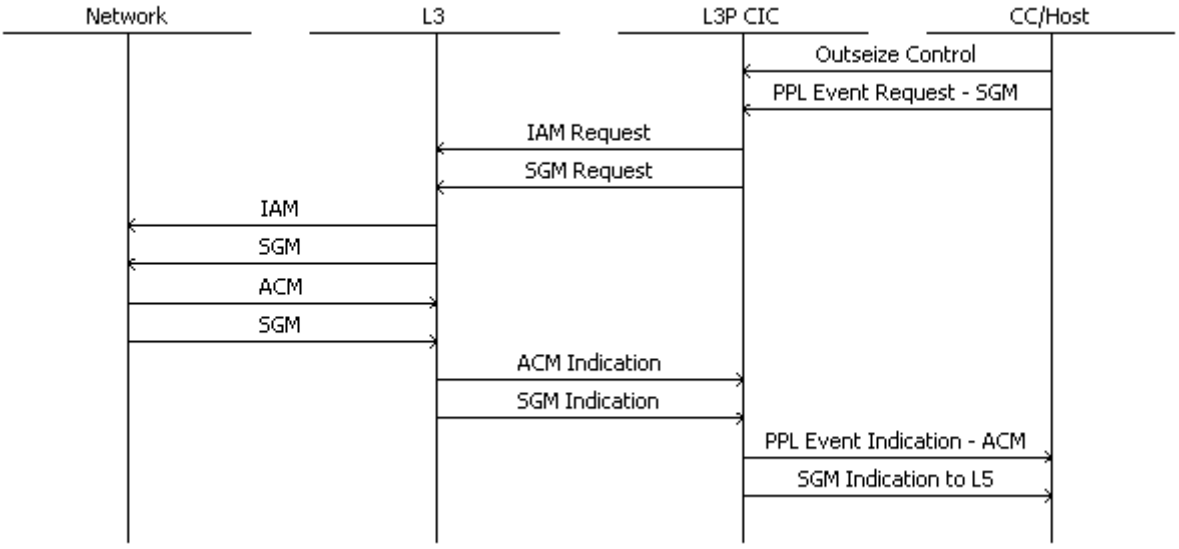
Incoming Call with Segmentation

The following call flow describes an incoming call with Segmentation. The SGM message is received after the IAM at the SSC component. Once the SGM is received, the RFS and the SGM message are sent to the host. (Alternatively, the SGM message and IAM can be concatenated and a single RFS can be sent to the host if the Configuration Byte 0x02 in the SSC component 0x85 is changed to 0x00.)



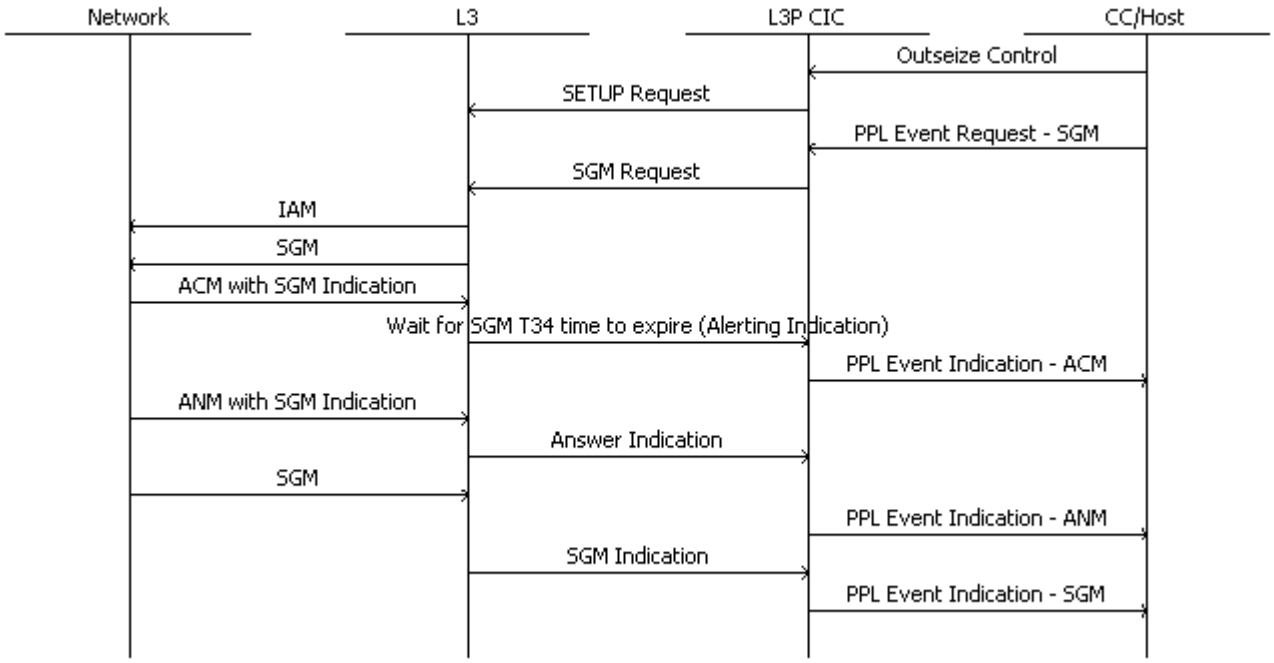
Outgoing Call with Segmentation in IAM and Backward Messages

When sending a large Outseize (> 272) bytes, the message is segmented by the SSC component. The following call flow shows a typical scenario for this process.

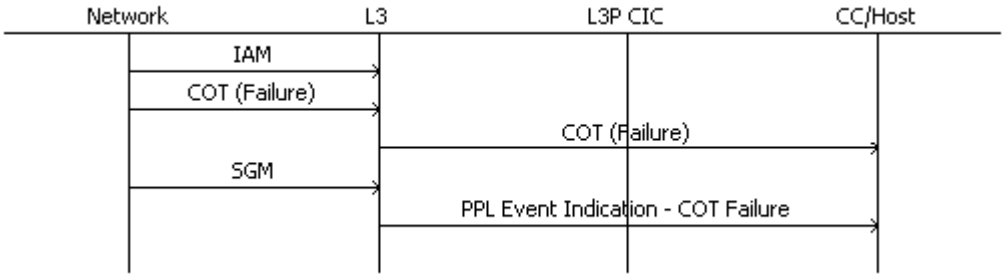


Incoming Segmentation Timeout

This call flow describes a scenario wherein the ACM message indicates that an SGM is to follow. However, if the SGM is not received until the expiry of T34 (typically 2 s), the ACM is forwarded to the Host. Any SGM message that follows is discarded.



Segmentation with Continuity Failure



ANSI ISUP Segmentation

The ANSI ISUP Segmentation feature allows you to select different interpretation and segmentation handling for the forward indicator E bit in the Initial Address Message (IAM) for ANSI segmentation.

The ANSI ISUP specification T1.113 - 1995 defines the E bit of the forward indicator in Initial Address Message (IAM) as the IAM Segmentation Indicator. If the E bit is set in the IAM, the receiving exchange will start timer T36 and wait for the Information (INF) message. During this time, the Address Complete Message (ACM) will not be sent back even when requested by the host call control.

However, the ANSI ISUP specification T1.113 - 2000 defines the E bit of the forward indicator as a spare. In this case, ISUP implementation may run into the problem when receiving an IAM with the forward indicator E bit set, and call control requests such as Connect AB, PPL Event Requests of ACM from the host call control are received before timer T36 expires. These messages will be positively acknowledged, but the ACM will not be sent to the network. Eventually, the call fails when timer T7 expires from the transmission end.

The ISUP CPC PPL Component (0x0012) provides the ANSI ISUP Segmentation (0xD4) configuration byte to define whether the forward indicator E bit in the IAM should be interpreted as a segmentation indicator. When this configuration byte is set, timer T36 will be started for the INF message, and Call Control request for the ACM will not be processed if received before timer T36 expires. If this configuration byte is not set, the forward indicator E bit is considered as a spare, and the ACM can be sent out immediately, and there is no waiting period for the INF message.

ISUP Continuity Check

Overview

This feature provides the CSP ANSI and ITU SS7 variants with a means to perform a continuity check to verify the integrity of the voice circuit between two exchanges before use of the circuit.

This section provides general information applicable to both ANSI and ITU variants then provides specific information on each as follows:

ITU ISUP Continuity (4-32)

ANSI ISUP Continuity (4-35)

Specifications

The requirements for the Continuity Check on the outgoing circuit are provided in the following recommendations for the International Telecommunication Union (ITU):

- ITU-T Q.724
- ITU-T Q.761 Functional Description of the ISDN User Part of Signalling System No. 7
- ITU-T Q.762 General Function of Messages and Signals of the ISDN User Part of Signalling System No. 7
- ITU-T Q.763 Formats and Codes of the ISDN User Part of Signalling System No. 7
- ITU-T Q.764 Signalling System No. 7 - ISDN User Part Signalling Procedures

The requirements for the Continuity Check on the outgoing circuit for the North American variant are provided in the following recommendations:

- ANSI T1.113.4 Signalling System No. 7 (SS7) - Integrated Services Digital Network (ISDN) User Part.

Required DSP Equipment

A Call Progress Tone (CPT) transmitter, and a Call Progress Analysis (CPA) receiver of appropriate PCM encoding are required in order to perform outgoing continuity checks.

Implementation

The ISUP part of the SS7 stack (ANSI and ITU-T) has provisions for a continuity check on the outgoing side of the circuit. A continuity check is initiated with either the Continuity Check Request message to the network side, or the setting of the continuity check indicator bits in the Nature of Connection Indicators mandatory parameter field within the IAM message.

These initial messages allow the network side of the circuit to loop back the chosen circuit, so that continuity testing can be done on that circuit. The CCO component within ISUP executes a continuity test to determine if the circuit to be used has continuity before it is used.

A COT message containing the Continuity Indication is sent to the network side when testing is complete. This indication determines whether continuity exists on the circuit or whether more continuity testing is required.

If a Continuity Recheck procedure is running (CRO for ANSI, CRCS for ITU), outgoing calls on the circuit are rejected with an error code of 0x0A.

ITU ISUP Continuity

Configuration

This section describes configuration required for the ITU ISUP Continuity Check procedure.

Continuity Check

The Continuity Check feature is disabled by default. To enable it, change the following L3P CIC PPL Configuration Byte to 0x04 using the *PPL Configure* message:

- ITU - Configuration Byte 135

Continuity Recheck

The Continuity Recheck procedure is enabled by default. The default number of retries is 2. These are configurable by modifying the following PPL Configuration Bytes using the *PPL Configure* message:

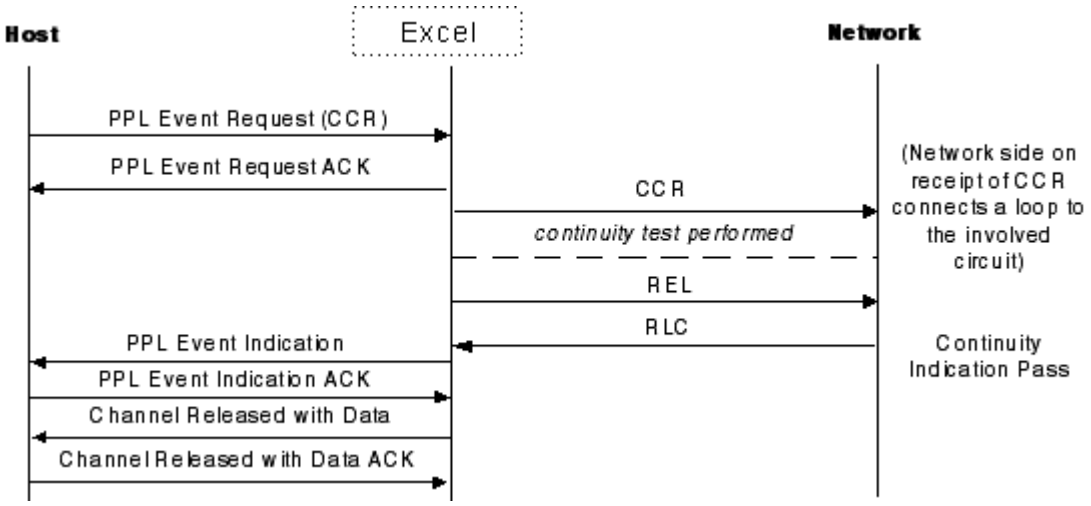
- ITU - ISUP CRCS (0x81)
  - Change Configuration Byte 10 to 0x00 to disable
  - Change Configuration Byte 11 for number of retries

To send a CCR message to the CSP, send the *PPL Event Request* message with an event value of 0x65. To stop the sending of the CCR message, send event value 0x64.

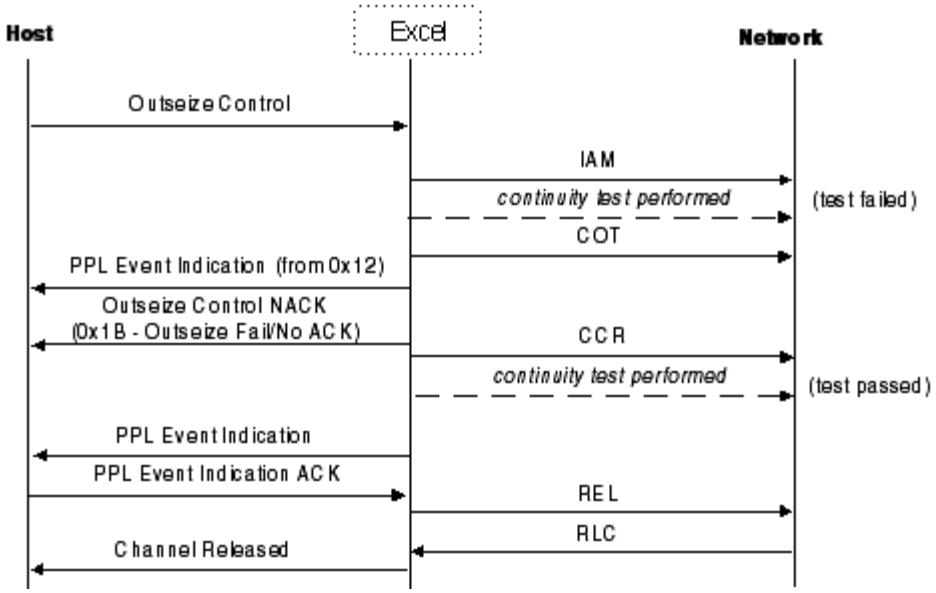
Call Flows

CCR

The call flow for a CCR test on a circuit.



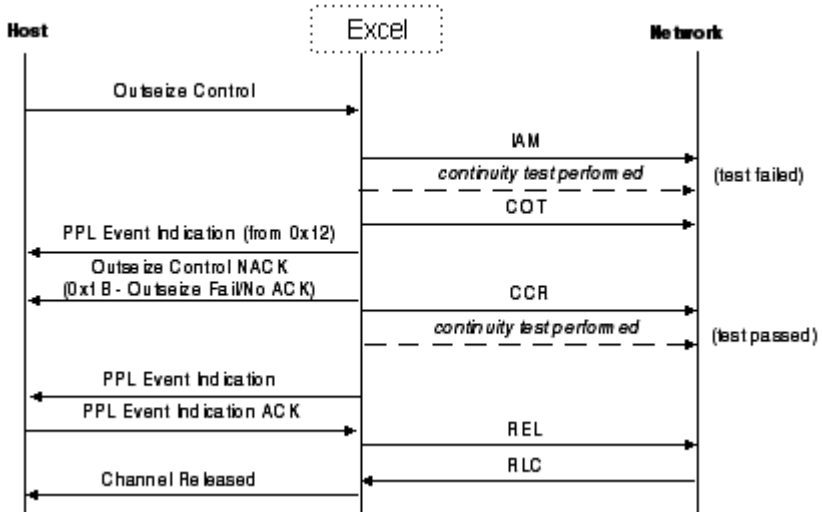
IAM with a Continuity Request



IAM with a Continuity Request and a PPL Event Indications sent to the host

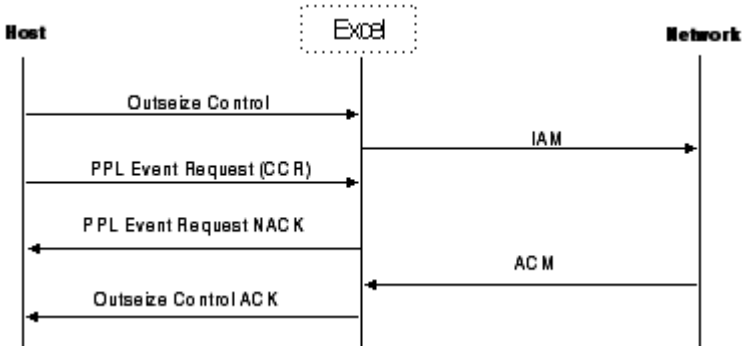
If the COT test fails, it is retried until it passes or the number of retries (indicated by Configuration Byte 11 of the ISUP CRCS component (0x81)) is reached.

.



CCR Sent to Active Channel

The call flow for an error condition where a CCR is sent on an active channel.



ANSI ISUP Continuity

Introduction

This section explains the various aspects of ISUP continuity procedure for ANSI as implemented in the CSP. It includes:

- message flows between exchanges
- the involvement of host application in each scenario
- the timer involved and various configurable parameters

ISUP Continuity Check

Continuity check is performed to verify the integrity of the voice circuit between two exchanges before use of the circuit. An exchange can initiate a continuity check on a circuit intended to be used for a call by one of the following ways:

1. Sending an IAM (Initial Address Message) message with a Nature of Connection parameter having continuity check indicator bits set to "required on this circuit".
2. Sending a CCR (Continuity Check Request) on the circuit which fails the initial continuity test mentioned in 1.
3. Sending a CRM (Circuit Reservation Message) with the Nature of Connection parameter having continuity check indicator set to “required on this circuit”

The exchange does the following when it receives these messages:

1. Does a “loopback” on the circuit under test.
2. Starts timer T8 (15 s). Expiration of T8 timer causes the connection to be cleared with cause 41.

The originating Exchange tests the continuity of the circuit by, attaching a transceiver on the circuit under test, sending a tone and verifying that the same tone is received via the loop back.

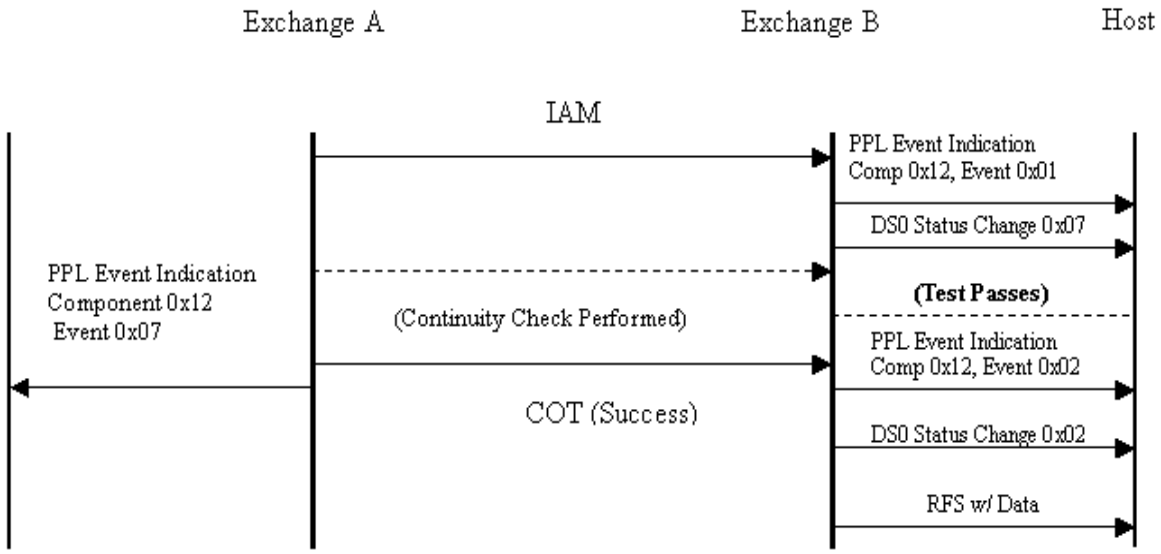
The continuity test result is propagated to the destination exchange by an ISUP message called COT with the continuity indicator mandatory parameter set to "failed" or "passed".

Example - COT Success

The call flow below shows IAM from Exchange A with nature of connection parameter set to continuity required on this circuit followed by a COT (successful) message.

Exchange A’s CPC component (0x12) sends a PPL event indication of Continuity Check Outgoing Success (0x07) to the host.

At the receiving side, the host receives a PPL event indication of COT indication in IAM (event 0x02) followed by a DS0 Status Change of 0x07 (Maintenance Loopback), Exchange B's CPC component (0x12) starts timer T8 and goes into wait for COT state after receiving IAM with nature of connection parameter set to continuity required on this circuit. On receiving the COT (Success) message, the timer T8 is stopped, and the host is informed by sending a PPL event indication of COT success (event 0x02) followed by a DS0 Status Change of 0x02 (In-service) and subsequently the RFS with Data.



Components Involved

CPC (0x12)

Timers Involved:

T8: Wait for COT message

Example - COT Failure

The call flow below shows the IAM from Exchange A with nature of connection parameter set to continuity required on this circuit followed by a COT (fail) message.

Exchange A's CPC component (0x12) sends a PPL event indication of Continuity Check Outgoing Failure (0x08) to the host and starts the CRO (Continuity Recheck Outgoing) procedure. The host will receive an Outseize Control NACK of 0x1b to its *Outseize Control* message, and subsequently a Channel Released message (0x49).

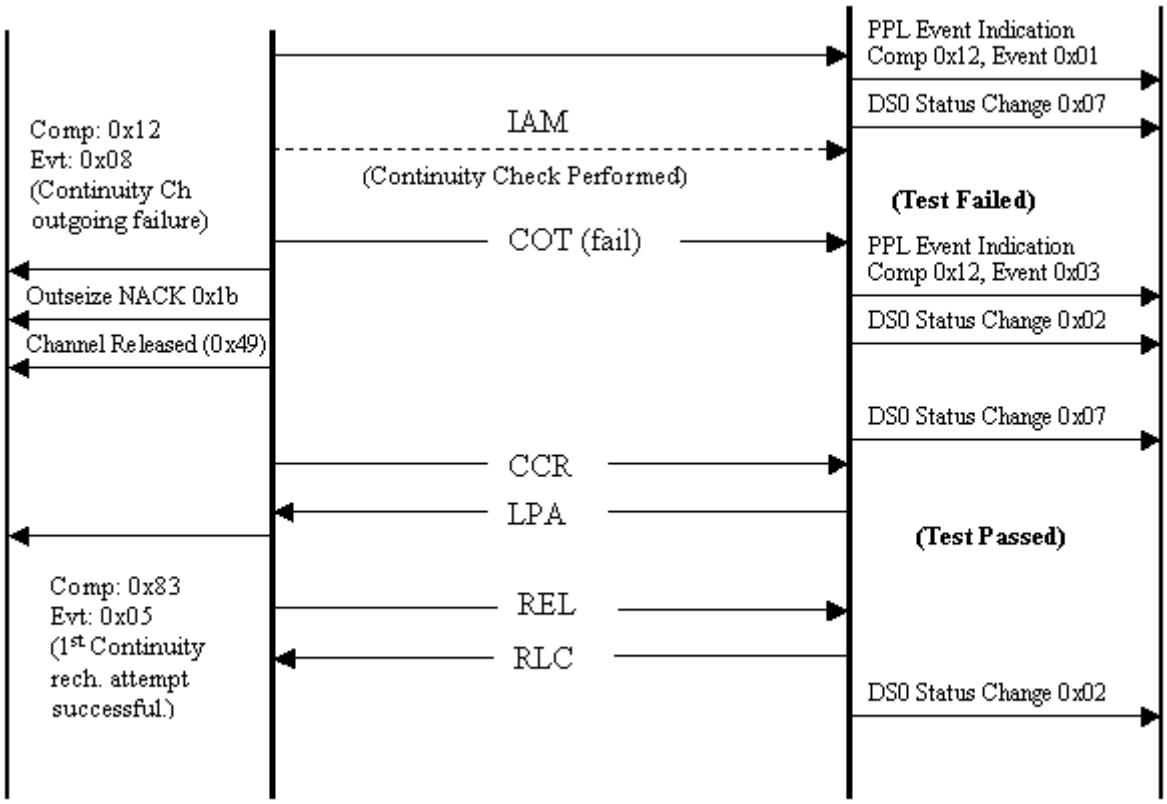
Important! The host should not consider this channel idle. Attempts to send subsequent Outseize Control messages will result in a 0x0a NACK (Continuity Recheck Running)

The CRO component (0x83) takes over from this point and automatically sends a CCR (Continuity Check Required) message to Exchange B.

This time the continuity test passed and the CCR is followed by REL message. The host (Host A) is informed of this by a PPL event indication of *1st Continuity Recheck attempt successful* (0x05) from the CRO component (0x83).

On the inbound leg, Exchange B's CPC component sends a PPL event indication of COT failure (0x03) to the host on receiving the COT (failed) message and starts the CRI (Continuity Recheck Incoming) procedure. The CRI component (0x15) takes over from this point. The CRI component, upon receiving the CCR message, applies a loopback on the circuit sends LPA (Loopback Acknowledgement) and waits for REL or COT message.

A REL message implies that the continuity check was successful and the circuit is put in idle state. A COT message implies that the COT failed. This causes the CRI component to go in wait for CCR state and the whole procedure is repeated.



Components involved:

- CPC(0x12)
- CRI(0x15)
- CRO(0x83)

Timers involved:

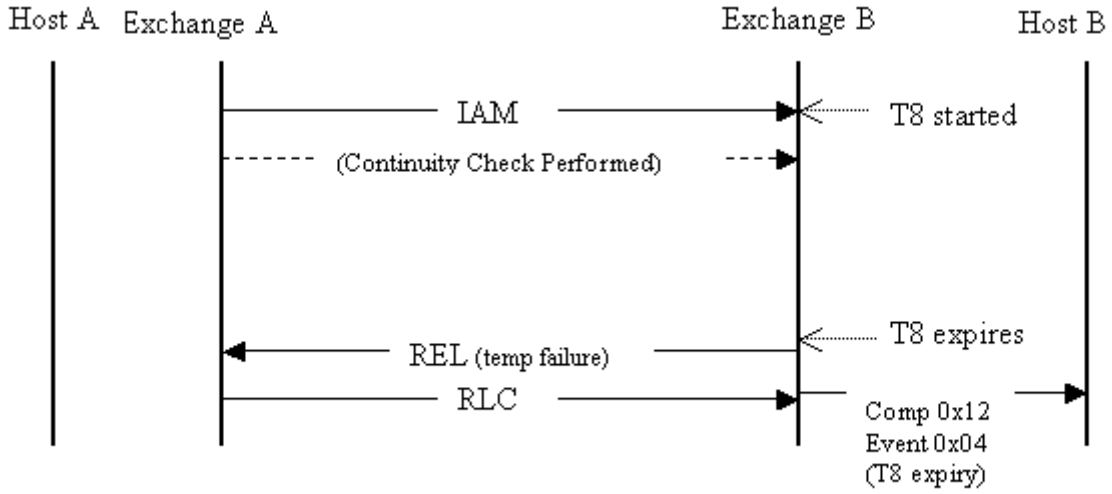
- CRI:
  - T-CCR: Wait for CCR message
  - T-34: Wait for COT report or REL message
  - T-27: Wait for continuity Recheck (second CCR message)
- CPC:
  - T8: Wait for COT message
- CRO:
  - T25: Wait before starting the repeat continuity check after continuity failed

Notes:

- Upon a successful continuity test after CCR message, the originating exchange does not send a COT (pass) message but a release message.
- A Call Progress Tone (CPT) transmitter and a Call Progress Analysis (CPA) receiver of appropriate PCM encoding are required in order to perform outgoing continuity checks.
- Host B is intimated about the incoming call before or after the COT result depending upon the value of config byte 0x09 of the CPC component 0x12. See Continuity check on previous circuit for more information.

Example - T8 Expiration

When, due to some reason, Exchange A fails to send a COT report to Exchange B within T8 seconds (default value 15 seconds), Exchange B sends a REL message to Exchange A with a cause of temporary failure as shown in the call flow below.



Continuity Check Request (CCR)

A COT fail message followed by CCR messages (after T25 of the COT-fail) is sent to Exchange B in case the continuity test fails after the first CCR message. CCR message is sent repeatedly in such cases (repeated COT failures) after every T26 seconds (timer value).

The Repeated continuity check will only be finished when continuity is detected (that is, continuity test passes) or manual intervention occurs.

Example - Repeated COT Failure

As shown in the call flow below, in the case of repeated failures the CRO component repeatedly sends CCR to the destination exchange. The first recheck result is indicated to the host by the PPL event indication of “1st Continuity Recheck attempt failure” (0x03) from the CRO component (0x83) and the subsequent failures are indicated by “Continuity Recheck attempt failure” (0x04).

At Exchange A, the host will receive an *Outseize Control* NACK of 0x1b to its *Outseize Control* message (0x002C), and subsequently a Channel Released (0x0049) message.

The host should not consider this channel idle. Attempts to send subsequent Outseize Control messages will result in a 0x0a NACK (Continuity Recheck Running). Upon reception of “Continuity Recheck attempt failure” (0x04) indication, the host may choose to stop the Recheck procedures by sending a “Manual Stop” PPL Event Request of 0x40 to component L3PCIC (0x000f).

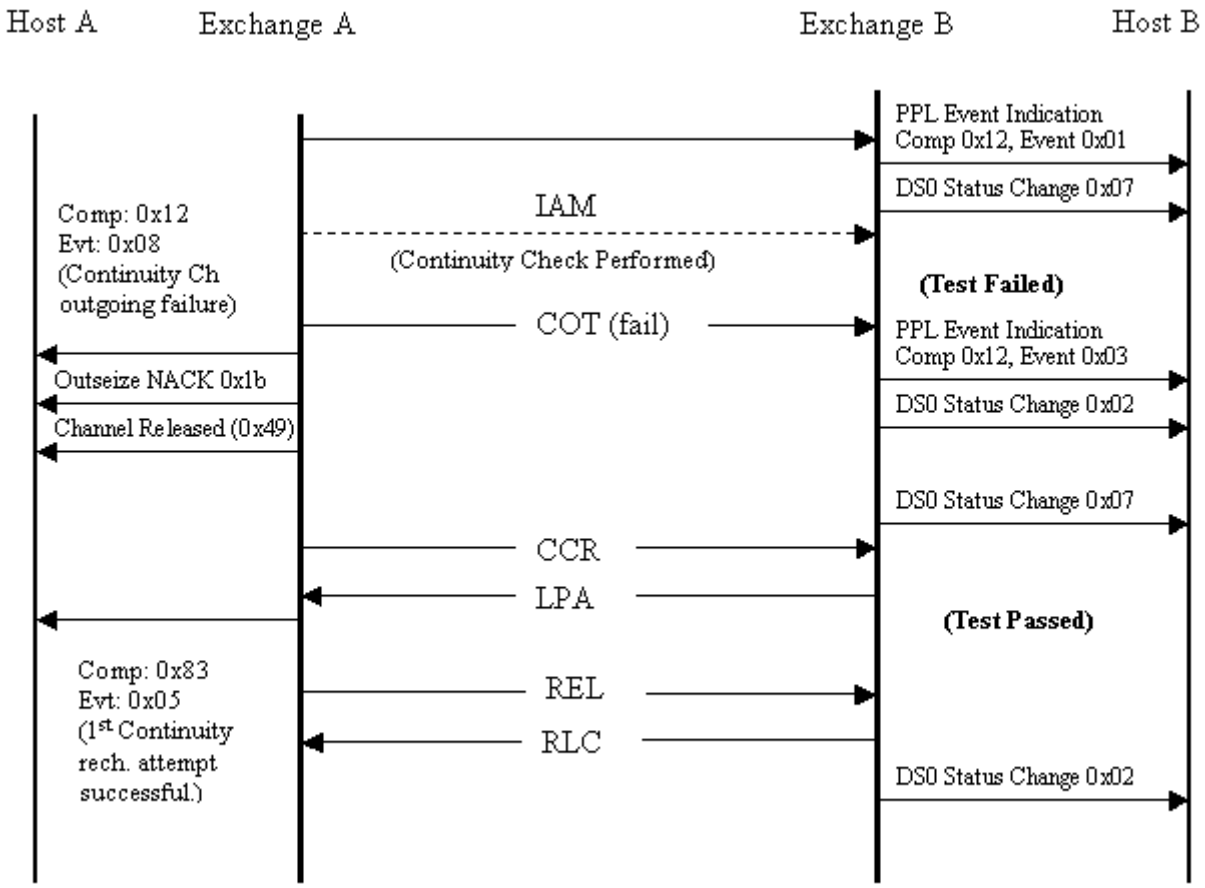
Exchange A will send a REL message to clear the circuit, and the host will receive a PPL Event Request of 0x01 “Subsequent Continuity Recheck attempt successful” (0x01) from component 0x0083 (CRO). This request is slightly misleading since the recheck was technically not successful in terms of a check tone. The host must recognize that this event is in response to its request to stop the proceedings.

At Exchange B, a success in one of the subsequent attempts is informed by a PPL event indication of “Subsequent Continuity Recheck attempt successful” (0x01). During this time, the inbound circuit will be placed in a Maintenance Loopback state indicated by a DS0 Status Change of 0x07. A DS0 Status Change of 0x02 will notify the host at Exchange B of the completion of continuity checks.

To avoid getting into an infinite loop in case of repeated failures, the CSP gives an option of limiting the continuity checks.

The default behavior of the CSP is to initiate a CCR message after a failed COT and try it four times in case the subsequent continuity checks fail. You can modify the default of 4 retries on the CSP by changing the CRO configuration byte 0x0b value from 2 (two mandatory + 2 default configuration byte = 2 total default) and retaining the default value of configuration byte 0x0a as 1 (which means that stopping of repeated continuity checks is enabled).

Changing configuration byte 0x0a to 0x00 means that the continuity check retries will only stop when continuity is detected (that is continuity test passes) or manual intervention occurs. In this case configuration byte 0x0b is not considered.



Components involved:

- CPC(0x12)
- CRI(0x15)
- CRO(0x83)

Timers involved:

- CRI:
  - T-CCR: Wait for CCR message
  - T-34: Wait for COT report or REL message
  - T-27: Wait for continuity Recheck (second CCR message)
- CPC:

  - T8: Wait for COT message
- CRO:
  - T25: Wait before starting the repeat continuity check after continuity failed
  - T26: Wait before doing repeated continuity check after subsequent continuity failures

Configuration Bytes

CRO: Continuity Recheck Outgoing (0x83)		
Byte	Usage Description	Options (* default)
0x0A	Stop Continuous Continuity Recheck	0x00 = No 0x-1 = Yes (default)
0x0B	Number of Times to Recheck	2 (default)

Continuity Recheck

In case of continuity recheck, the CSP, as recommended by ANSI, sends a COT success report to the destination exchange (and to the CRO component) only when the continuity test is stopped by the CRO component (this is typically done by manual intervention).

At the destination exchange the COT-pass message is misleading in such cases (since the continuity has not actually passed). In view of this, the CSP does not wait for a COT pass message in the CRI component. It just waits for REL which implicitly conveys a successful COT report. Any COT report (fail or pass) coming from the originating exchange is considered as fail at the destination exchange CRI and the CSP keeps waiting for release or timer expiration.

Continuity on Previous Circuit

When an exchange receives an IAM-continuity check required message, it can do the following:

- Choose to wait for the COT-success result before propagating the IAM.
- Choose to propagate an incoming "IAM with continuity required", without waiting for the COT result. The outgoing IAM in this case will have the NOC parameter set to continuity check on a previous circuit.

The CSP makes this decision based on the configuration byte 0x09 in the CPC component (0x12). Setting this configuration byte to 0x01 allows the CSP to send the RFS to the host upon receiving an IAM with continuity request. The host can then propagate this IAM with NoC set to continuity check on a previous circuit. If the configuration byte is set to 0x00, the RFS is sent to the host only after a COT- success is received.

Both these cases are shown in the call flows in corresponding example.

When changing the configuration byte 0x09 in component 0x12 to 0x01, configuration byte 0x2b in component 0x0f must also be set to 0x01.

The host should not try to send any call processing API on the circuit where incoming Continuity Check is being performed, until continuity check results (of pass) are received in the form of a PPL Event Indication.

Exchange B will propagate the COT coming from Exchange A only if it is COT –passed.

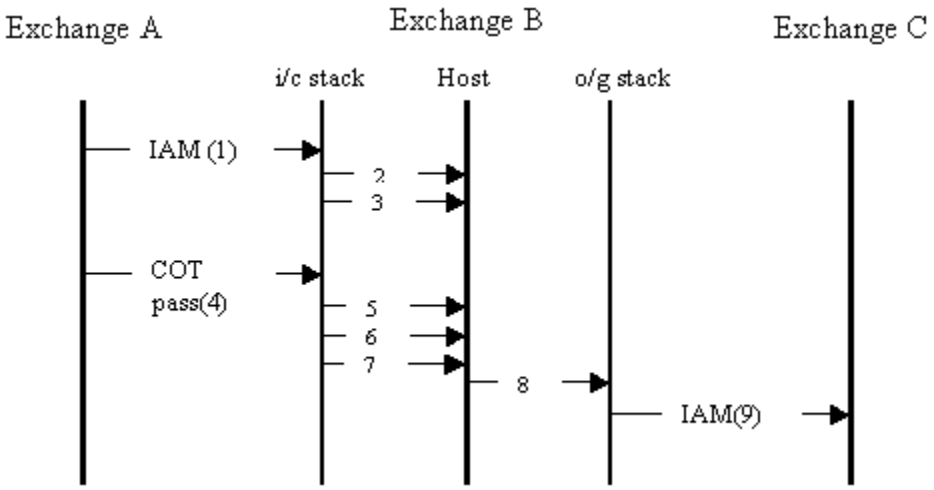
In case, Exchange B receives COT-fail from Exchange A, it is not propagated to Exchange C and Exchange C clears the call upon the expiration of timer T8.

Example - Continuity Check on Previous Circuit

In the call flow below, Exchange B waits for COT results before propagating the IAM.

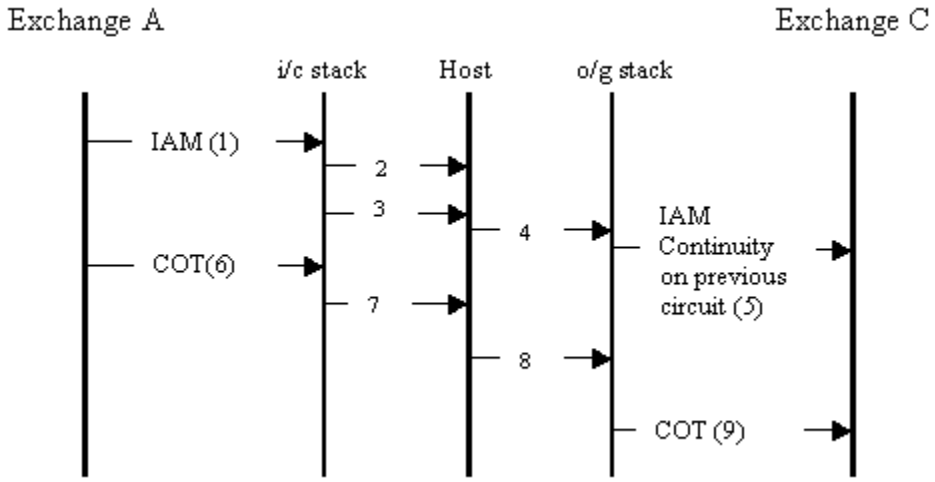
On component 0x12, configuration byte 0x09 is set to 0x00, send RFS after continuity is performed.

On component 0x0F, configuration byte 0x2B is set to 0x00, send DS0 status change of Channel Out of Service (Loopback) to the host.



Call Outs

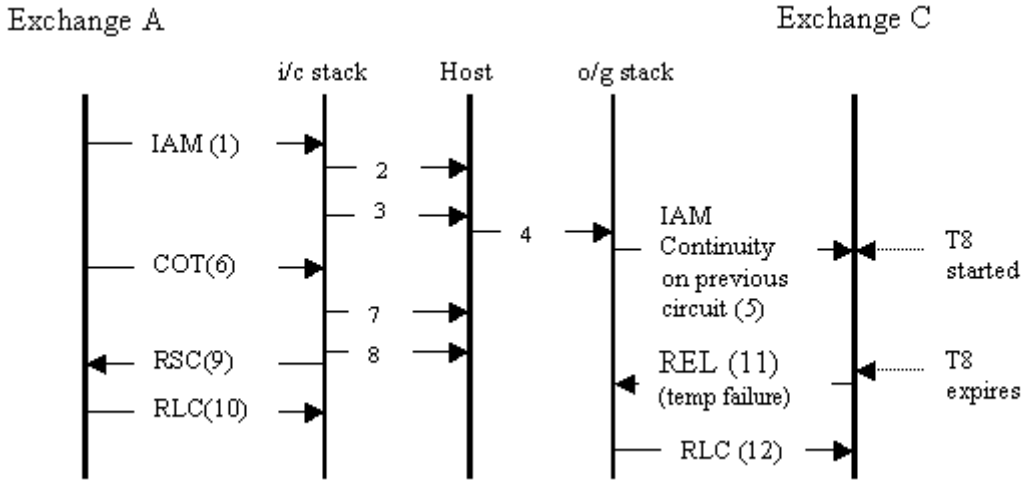
1. The IAM with continuity check is required on this circuit.
2. The PPL Event Indication of 0x01 from component 0x12 (IAM received with the Continuity Check indicator set.)
3. The DS0 status change of OOS (maintenance loopback) is sent only if configuration byte 0x2B of component 0x0F is set to 0x00.
4. The ISUO message of COT success is sent.
5. The PPL Event Indication 0x02 from component 0x12 (COT success from CPC).
6. DS0 status change of in-service.
7. RFS (Request For Service) from L4 to host.
8. *Outseize Control* message from host to L4.
9. IAM (continuity check not required).



Call Outs

1. IAM (continuity check on this circuit)
2. PPL Event Indication of 0x01 from component 0x12 (IAM received with Continuity Check indicator set.)
3. RFS (Request for Service) from L4 to host. This is sent without waiting for a COT success message as configuration byte 0x09 is set to 0x01
4. *Outseize Control* message.
5. IAM (continuity check on previous circuit).
6. COT success.
7. PPL Event Indication of 0x02 from component 0x12 (COT success).
8. PPL Event Request 0x0D from component 0x0F (COT success).
9. COT success propagated.

In the following call flow, Exchange B propagates IAM with continuity test on previous circuit. COT failure is not propagated to the destination exchange.



Call Outs

1. IAM (continuity check on this circuit).
2. PPL Event Indication of 0x01 from component 0x12 (IAM received with Continuity Check indicator set).
3. RFS (Request for Service) from L4 to host. This message is sent without waiting for a COT success message as configuration byte 0x09 is set to 0x01.
4. Outseize Control message.
5. IAM (continuity check on previous circuit).
6. COT failure.
7. PPL Event Indication of COT fail.
8. Channel release (0x0049).
9. RSC (Reset Circuit).
10. RLC (Release Complete).
11. REL (temp failure) on T8 expiration on Exchange C.
12. RLC (Release Complete).

Components involved:

- CPC (0x12)
- L3P CIC (0x0F)

Timers Involved

- CPC
- T8: Wait for COT message

Configuration Bytes Involved

CPC: Call Processing Control (0x12)		
Byte	Usage Description	Options (* default)
0x09	Send RFS to the host after incoming continuity result is received. When setting this byte to 0x01, the host should not try to send any call processing API messages on the circuit where incoming Continuity Check is performed - until the continuity check results are received in the form of a PPL Event Indication.	0x00* = Send RFS after continuity has been performed (default) 0x01 = Send RFS before continuity has been performed.

L3P CIC (0x0F)		
Byte	Usage Description	Options (* default)
0x2B	Send RFS to host before incoming continuity result is received and connect the loopback without OOS. In this case, the host should not try to send any other call processing messages on the circuit where an incoming continuity check is being performed - until the continuity check results are received in the form of PPL Event Indication.	0x00* = Send DS0 status change of Channel Out of Service (Loopback) to the host. 0x01 = Do not send DS0 status change of Channel Out of Service (Loopback) to the host.

Incoming and Outgoing Continuity

In some cases, both the incoming and outgoing legs of a call have continuity required on the circuit as shown in the call flow in the example. See *Example - Incoming and Outgoing Continuity (4-48)*.

In such cases Exchange B can determine if it wants to wait for a COT success message from Exchange A before sending a COT success message to Exchange C.

In case Exchange B wants to wait for incoming COT report before sending an outgoing COT success, Exchange C will get a combined COT success for the continuity check required for this circuit and the previous circuit.

1. Exchange C receives a COT success message: Means that the continuity of this circuit (between B and C) AND all the preceding circuits for which the continuity test was in progress (like between A and B) has been verified.
2. Exchange C receives a COT failure message: Means that continuity on this circuit (between B and C) has failed
3. Exchange C does not receive a COT message: Means that a previous circuit (like between A and B) might have failed the continuity test. Exchange C does not need to know which circuit failed as long as it is not its own (in which case it is required to start a continuity recheck loop back)

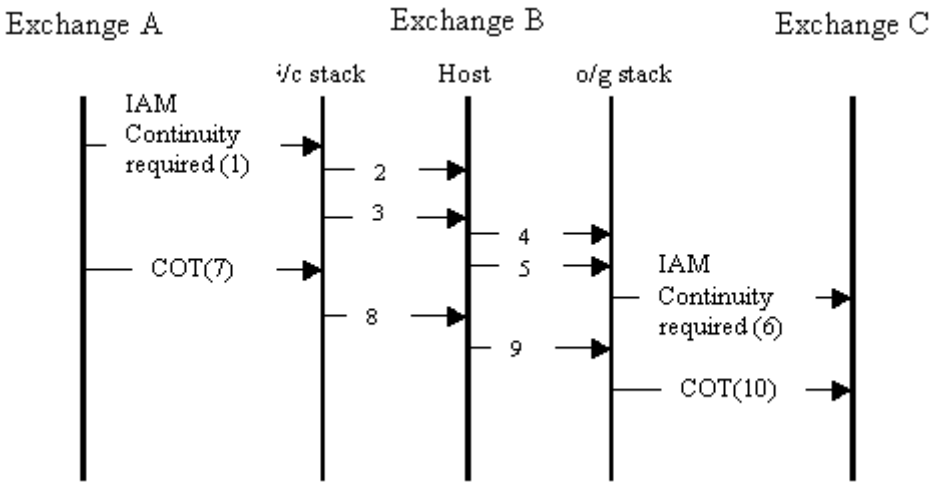
If the Exchange B chooses not to wait for the continuity success message, Exchange B will send a COT success or failed message as soon as the continuity test between B and C is done and the call will proceed beyond Exchange C (in case the COT was successful). There can be two cases here:

1. If Continuity test passes between Exchanges B and C but fails between Exchanges A and B, then Exchange B incoming side will release the call and sends a PPL Event Indication of COT failure to the host. On the outgoing side (between B and C), the onus lies with the host to release the call and free the outgoing circuit.
2. If Continuity test passes between Exchanges B and C and between Exchanges A and B, Exchange B should not propagate the COT message received to the outgoing side and should let the call complete.

Example - Incoming and Outgoing Continuity

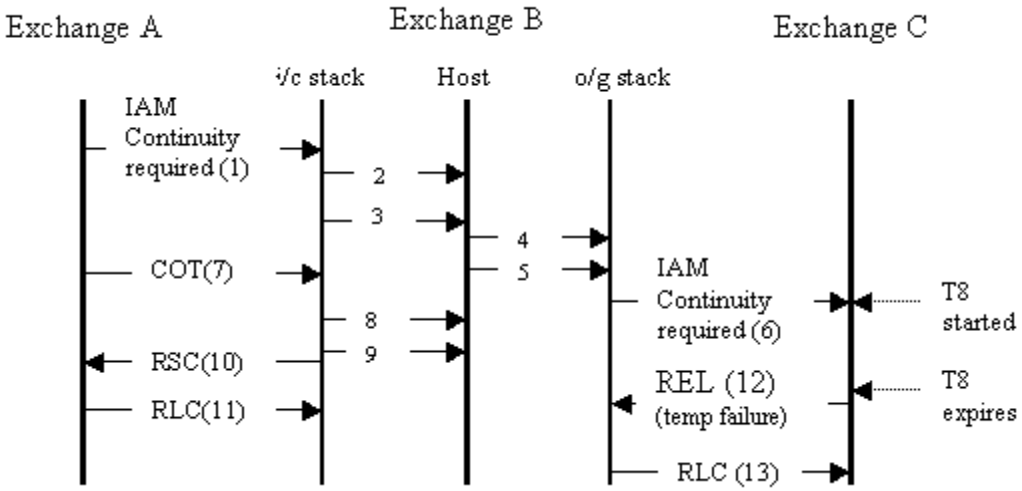
The host sends a PPL event request of 0x0C to component L3P CIC component (0x0F) to make the CSP (Exchange B) wait for the incoming COT report (from Exchange A) before sending COT success to Exchange C.

At Exchange B the host passes information of the incoming COT report to the outgoing L3P (and hence CPC) component by sending a PPL event request of 0x0d to the L3P component. This request conveys an incoming COT success.



Exchange B propagates COT success only when both inbound and outbound continuity tests pass.

1. IAM (continuity check on this circuit)
2. PPL event indication of 0x01 from component 0x12 (IAM received with Continuity Check indicator set)
3. RFS (Request For Service) from the CSP to the host. This is sent without waiting for a COT success message as config byte 0x09 is set to 0x01
4. *Outseize Control* message
5. PPL Event Request 0x0c from component 0x0f to indicate CPC to wait for incoming COT report when continuity check is being performed on previous circuit AND this circuit (i.e. both incoming and outgoing continuity are being performed).
6. IAM (continuity check on this circuit)
7. COT success
8. PPL event indication of 0x02 from component 0x12 (COT success)
9. PPL event request 0x0d from component 0x0f (COT success).
10. COT success propagated. This is sent only after a successful continuity test is done on circuits between Exchanges B & C and between A & B.



Exchange B does not propagate COT when the inbound COT fails.

1. IAM (continuity check on this circuit)
2. PPL event indication of 0x01 from component 0x12 (IAM received with Continuity Check indicator set)
3. RFS (Request For Service) from L4 to host. This is sent without waiting for a COT success message as config byte 0x09 is set to 0x01
4. *Outseize Control* message
5. PPL event Request 0x0c from component 0x0f to indicate CPC to wait for incoming COT report when continuity check is being performed on previous circuit AND this circuit (i.e. both incoming and outgoing continuity are being performed).
6. IAM (continuity check on this circuit)
7. COT Failure
8. PPL event indication of 0x03 from component 0x12 (COT failure)
9. Channel released (0x0049)
10. RSC (Reset Circuit)
11. RLC (Release Complete)
12. REL (temp failure) on T8 expiration on Exchange C
13. RLC (Release Complete)

Components Involved

- CPC (0x0012)
- L3P CIC (0x000F)

Timers Involved

- CPC
  - T8 Wait for COT message

PPL Information for Continuity

The following summarizes the PPL information used for ANSI ISUP Continuity. Refer to *SS7 PPL Information* for the details of each component listed below.

Event Requests

- L3PCIC (0x000F), 0x0C, 0x0D

Event Indications

- Call Processing Control (0x12), 0x01-0x08
- Continuity Recheck Outgoing (0x83), 0x01-0x05

Configuration Bytes

- Call Processing Control (0x12), 0x09
- L3PCIC (0x000F), 0x2B
- Continuity Recheck Outgoing (0x83), 0x0A, 0x0B

PPL Timers

- Call Processing Control (0x12), 0x01
- Continuity Check Outgoing (0x14), 0x18
- Continuity Recheck Incoming (0x15), 0x01-0x03
- Continuity Recheck Outgoing (0x83), 0x01, 0x19, 0x0A

Host Controlled Continuity

The continuity feature as implemented on the CSP may prove ineffective when the host wants to specifically take control of the continuity procedure. A typical example where this may be required is in situations where the Voice circuits and Signaling links do not reside on the same CSP or where the voice circuits being used for the call are virtual circuits. This section mentions the changes made in the ISUP layer to implement Host controlled Continuity Check procedure.

Host controlled continuity has been implemented in a way that it does not impact the existing COT procedure. A configuration byte 0xCC has been introduced in the CPC components of ISUP, which when left to its

default value of 00 will leave the existing continuity procedure as it is. When this configuration byte is changed to 01, the following changes in functionality take place:

- All COT processing in CPC is bypassed.
- All COT related inter-component messaging is bypassed.
- The COT, CCR and LPA (ANSI) messages received by SPRC are directly propagated to the Host via the CPC. This introduces new PPL event indications to the Host.
- The host will now be able to send PPL event requests for COT, CCR and LPA.

The onus of performing the continuity test and sending the COT (success/failure) to the CSP is on the host application. CSP has no control and no involvement in the continuity test procedure.

Circuit Status Query

Overview

The Circuit Query feature allows the host to query the status of a single circuit or a range of circuits using the *PPL Event Request* message. A Circuit Query Message (CQM) is sent to the remote CSP, which returns a Circuit Query Response (CQR) message containing the status of all the circuits. The local and remote circuit status for each circuit is sent to the host in a *PPL Event Indication* message.

Recovery Action for ANSI Only

For ANSI only, if the circuit is found in the invalid state by the CQM, a recovery action is taken by CQS. The host can configure the recovery action taken by the CSP using the PPL Configuration Bytes of the CQS component. Recovery actions taken by the CSP are as defined by the ANSI-ISUP T1.113 specification.

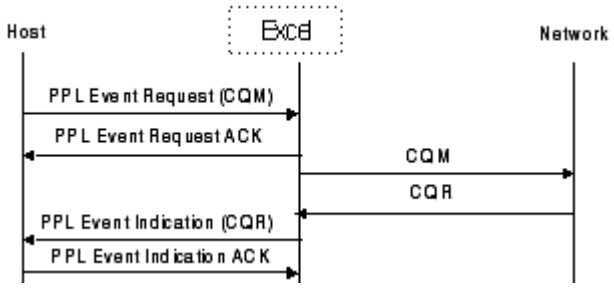
Implementation

When a CQM request is received from the host, the CSP sends a CQM message to the network. When a CQR is received from the network, the CSP sends a *PPL Event Indication* message to the host with an SS7 Data Parameters ICB (0x22) containing the status of the local and remote circuits. If the CQR indicates a maintenance state, the CSP takes appropriate action based on Table 2 in ANSI T1.113.4.

If a CQR is not received before the expiration of T28 (default =10 seconds), a *PPL Event Indication* of 0x01 is sent to the host indicating the error.

Call Flows

This call flow shows a CQM request from the host and a CQR response from the network.



Status Values

Local and Remote Status

The local and remote status are reported to the host in an SS7 Data Parameters ICB (0x22) in a *PPL Event Indication* message, where:

Data[0]	Local Status
Data[1]	Remote Status

One of the following values is reported for both Data[0] and Data[1].

0x00	Transient

Maintenance States

0x01	Spare (no interpretation)
0x02	Spare (no interpretation)
0x03	Unequipped
0x04	Incoming Circuit Busy, Active
0x05	Incoming Circuit Busy, Locally Blocked
0x06	Incoming Circuit Busy, Remotely Blocked
0x07	Incoming Circuit Busy, Locally and Remotely Blocked
0x08	Outgoing Circuit Busy, Active

Call Processing States

0x09	Outgoing Circuit Busy, Locally Blocked
0x0A	Outgoing Circuit Busy, Remotely Blocked
0x0B	Outgoing Circuit Busy, Locally and Remotely Blocked
0x0C	Idle
0x0D	Idle, Locally Blocked
0x0E	Idle, Remotely Blocked
0x0F	Idle, Locally and Remotely Blocked
0x10 to 0xFF	Spare

Circuit Validation Test and Response

Overview

This feature provides a method by which the host can request the CSP to send a Circuit Validation Test (CVT) message to the far end and process the Circuit Validation Response (CVR). The circuit validation test procedures follow the recommendations of ANSI-ISUP T1.113.

The CVT and CVR messages are used to test and ascertain the consistency in circuit translation data on both the ends of a circuit, typically during circuit turn-up procedures to make sure that the circuit characteristics match. Maintenance personnel may also execute these tests on a routine basis.

The CVT/CVR procedure may also be invoked when a User Part Unavailable message with a cause value of Remote User Part Inaccessible is received by MTP.

Implementation

To initiate a Circuit Validation Test, send a *PPL Event Request* of CVT (0x002F) to the L3P CIC component (0x000F). After formatting the CVT message, ISUP passes the message on to the CVT component, which passes it to MTP3, which passes it on to the destination.

The destination node collects the required circuit information and returns a CVR message to the originating CVS component, which sends a PPL event to the host. The table below shows the PPL event sent to the host depending on the CVR contents.

CVR Indicator	Circuit Group Characteristics	CIN Parameters	CLLI Parameters	Indication to Host	PPL Event
Success	Match	Match	Don’t Care	CVT Successful	0x01
Success	Match	No CIN	Don’t Care	CVT Successful	0x01
Success	No Match	Match	Don’t Care	CVT Successful	0x03*
Success	Don’t Care	CIN Mismatch	Don’t Care	CIN Mismatch Indication (near-end and far-end CIN sent to host)	0x02*

* Data passed in Raw SS7 Data Parameters ICB (0x22)

** Data passed in SS7 Unformatted Raw Parameters ICB (0x1F)

CVR Indicator	Circuit Group Characteristics	CIN Parameters	CLLI Parameters	Indication to Host	PPL Event
Success	Don't Care	CIN Not Supported	Don't Care	CVT successful	0x01
Failure	Don't Care	Don't Care	CLLI Exists	CVT Failure indication with CLLI	0x05*
Failure	Don't Care	Don't Care	CLLI Does Not Exist	CVT Failure indication with no CLLI	0x04
Failure	Don't Care	Don't Care	CLLI Not Supported	CVT Failure indication with no CLLI	0x04
Local Circuit Translation Unavailable					0x07
Received CVR Not Processed – Pass Through to Host					0x08**
CVR Not Received After Maximum CVT Retry					0x06

* Data passed in Raw SS7 Data Parameters ICB (0x22)

** Data passed in SS7 Unformatted Raw Parameters ICB (0x1F)

PPL Events

The PPL events are sent to the host in the *PPL Event Indication* message. Event data, if any, is sent in an ICB. The ICB subtype and data for those events that include data are shown below. Events 1, 4, 6, and 7 do not include data.

0x02	CVT Successful - CIN Mismatch ICB Subtype: SS7 Raw Data Parameters (0x22) Data[0–25] - Local CIN Data[26–51] - Remote CIN
0x03	CVT Successful - CGC Data Inconsistent ICB Subtype: SS7 Raw Data Parameters (0x22) Data[0] - Remote CGC Parameter Data[1–26] - Local CIN
0x05	CVT Failure - CLLI Received ICB Subtype: SS7 Raw Data Parameters (0x22) Data[0–10] - Remote CLLI

0x08	CVR Received - CVR in TLV Format ICB Subtype: Formatted SS7 Parameters (0x12) Data [0] - Message Type [CVR] Data [1] - Number of Parameters Data [2] - CVR Indicator Parameters Code Data [3] - CVR Indicator Length Data [4] - CVR Indicator Value Data [5] - CGC Parameters Code Data [6] - CGC Parameters Length Data [7] - CGC Parameters Value Data [8-n] - Optional Parameters [TLV]

User Part Unavailable Message Handling

A CVT may also be initiated when a User Part Unavailable (UPU) indication is received from the far-end. This is configurable with PPL Configuration Byte 14 of the L3P CIC component.

When a UPU message with an appropriate cause value is received from the far end, a CVT message is sent and timer T37 (typically 30 s) is started. On receipt of a CVR or any other ISUP message, the timer is stopped and normalcy is restored. On expiry of T37 without the receipt of any response from the remote end, the CVT procedure is restarted.

The CVR received from the far end will contain the mandatory Circuit Validation Response indicator and Circuit Group characteristic indicator parameters and may also contain other optional parameters (CLLI and CIN code). CLLI/CIN can be enabled/disabled as required and the CLLI/CIN code parameters are configurable.

On receiving a CVT, a CVR with an appropriate CVR indicator of Pass or Fail is sent, depending on the state of the CIC. The CVR contains the configured Circuit Group Characteristics (CGC) parameter and may also contain the optional CLLI if the test fails or CIN if the test passes. For undefined CICs, no optional parameters are sent.

On receipt of an MTP Status primitive with a cause of User part unavailability – Remote user inaccessible, depending on the configuration, the CVT may be generated automatically. On receipt of a

CVR or any other ISUP message, the CVT/CVR procedures are stopped and a *PPL Event Indication* of 0x04 is sent from SPRC to the host.

Error Condition

When the CVT component sends a CVT message to SPRC it starts a TCVT timer. If a CVR is not received before the expiration of the timer, a second CVT is sent. If the timer expires a second time, the CVT attempt is cancelled and a *PPL Event Indication* (0x06 - CVR Receive Failed after Re-attempt) is sent to the host.

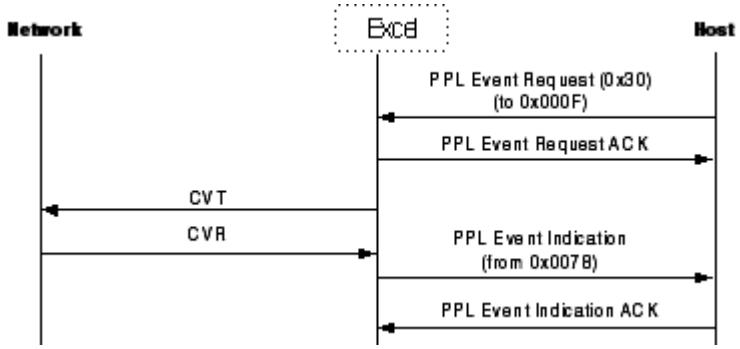
If a CVT procedure is started with Remote User Part Unavailable, the number of CVT re-attempts is determined by the value in Configuration Byte 8 of the CVS component. The default value is 8. The maximum value is 255.

Customization

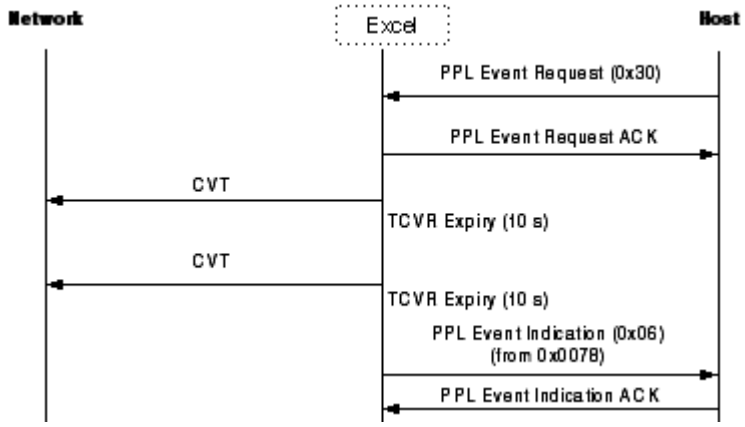
Use the PPL Configuration Bytes (using the *PPL Configure* message) of the CVT and CVR components to modify CVT/CVR configuration. The CIN, CLLI, and circuit group characteristics configuration should be same for a given circuit for both the CVT and CVR components. See *SS7 PPL Information* for the Configuration Byte values.

Circuit Validation Test Call Flows

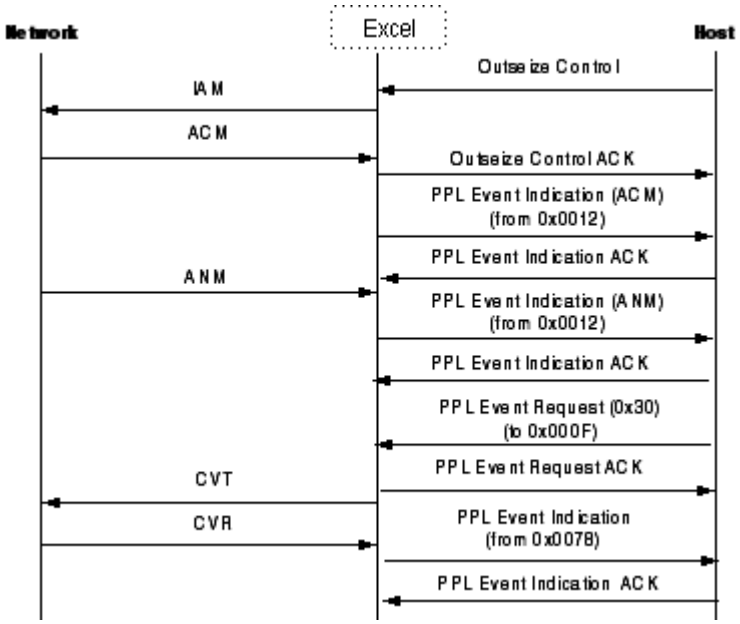
CVT to Network/CVR Received from Far-end



CVT to Network/No CVR from Far-end

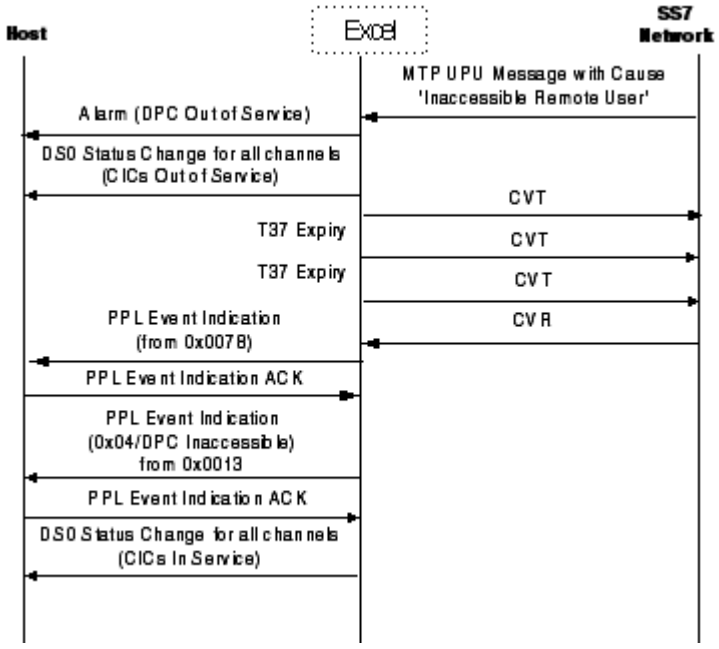


CVT to Network during Active Call



Remote User Part Unavailable

This call flow shows a CVT sent to the network as a result of a Remote User Part Unavailable indication received from the far-end.



Call Modification Messages

Overview

This section includes information about messages that involve requesting, rejecting, and completing call modifications.

CMR: Call Modification Request/Call Modification Reject

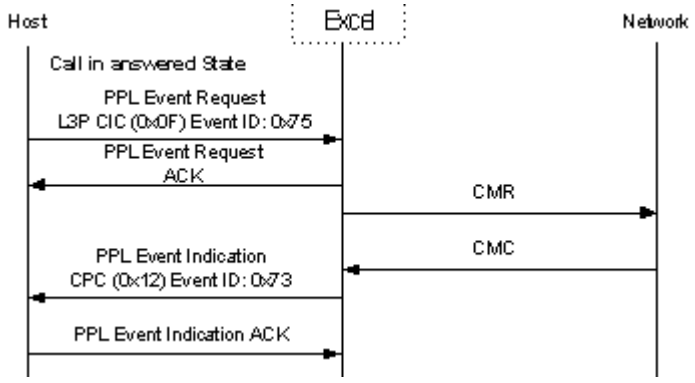
Message sent in either direction to indicate that the call modification request has been rejected.

CMC: Call Modification Complete

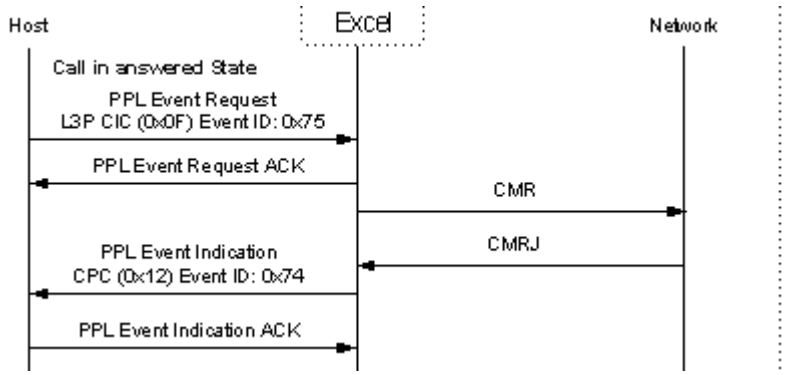
Message sent in either direction to indicate that the call modification request has been accepted

Call Flows

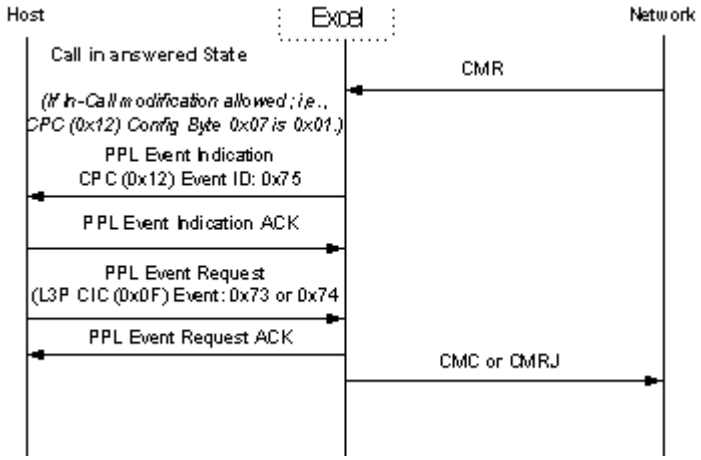
CMR from Host -- CMC from Network



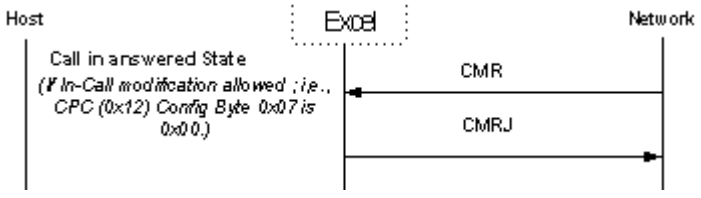
MR from Host -- CMRJ from Network



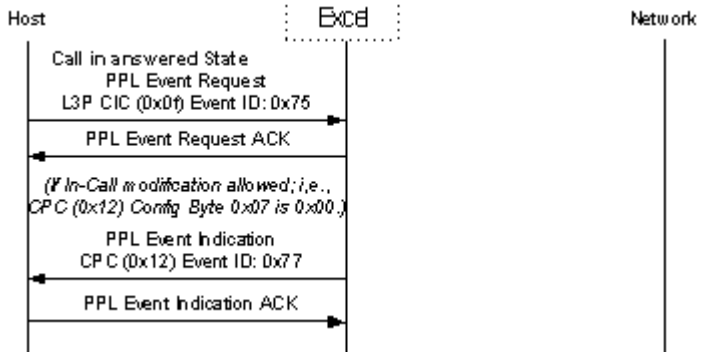
CMR from Network



MR from Network -- In-Call Modification Disabled



CMR from Host -- Local In-Call Modification Disabled



SS7 CIC Blocking Procedures

Overview

When a CIC changes between the out of service (OOS) and in service (INS) states, the SS7 card automatically initiates either a Circuit Blocking/Unblocking or a Circuit Reset for the circuit (or a Group Blocking/Unblocking or Reset if multiple circuits in the same circuit group transition at the same time). The two procedures for changing states are shown below and are controlled by PPL configuration.

Whenever an OOS to INS or the OOS is caused by a non-host initiated request (for example, a purge, or remote ISUP unavailable indication) the SS7 card attempts to send a Circuit Reset (RSC) or a Group Circuit Reset (GRS) (if multiple circuits are in transition at the same time). Also, if the host configures the CIC to send reset upon host-initiated OOS to INS transition, the SS7 card attempts to send a RSC or GRS.

> **Important!** The procedure for initial CIC In Service procedures is consistent with Type B operations as defined in ITU-TS Q.767 Section 4.4.1.

If the host configures the CIC to send blocking upon host initiated INS to OOS transition, the SS7 card attempts to send a Blocking (BLO) or Circuit Group Blocking (CGB) before taking the CIC out of service (this is the default).

When the host brings the CICs back into service, the SS7 card attempts to send an Unblocking (UBL) or Circuit Group Unblocking (CGU) before declaring the circuits as in-service.

For a CIC to start the OOS to INS process, the following three conditions must be met:

- The span on which the CIC resides must be in proper span alignment.
- MTP must consider the DPC as accessible.
- The host must send a *Service State Configure* message of in-service for the CIC.

When all the conditions are met, the SS7 card attempts to bring the CICs in-service. Upon successful completion of the ISUP Unblocking or Reset procedure, the CSP notifies the host by sending a *DS0 Status Change* message for each CIC, declaring it in-service. You can now use the CICs for calls.

A CIC purge is considered a non host-initiated OOS to INS transition. Because of this, any purge of a CIC initiates an outgoing Circuit Reset.

If the host wants to initiate a Circuit Group Reset (GRS) or Circuit Reset (RSC) it may do so by setting the L3P CIC PPL configuration byte 10 and then cycling the CICs OOS first and next INS through the use of the *Service State Configure* message. The host-initiated OOS to INS transition forces an outgoing GRS or RSC. This action is taken to force a reset of the entire CSP call processing stack, not just the ISUP portion of it.

A *PPL Event Request* message cannot generate an RSC or GRS. If a CIC is taken OOS from the host through the use of the *Service State Configure* message, the CIC is blocked if configured to do so and marked as Unequipped by ISUP. Any messages from the DPC are treated as messages for an Unequipped Circuit.

If a CIC circuit group is in service and a span alarm is detected, a Circuit Group Blocking message is automatically sent (or Circuit Group Hardware Blocking for ITU) for the concerned circuit group. The host receives a *DS0 Status Change* message indicating that all of the channels in the circuit group are OOS.

If a call was active, the CIC purges with the reason Span Status Dead (0x03). Upon return of proper span framing, a Circuit Group Unblocking is automatically sent (or Circuit Group Hardware Unblocking for ITU) for the concerned circuit group. Upon successful unblocking of the circuit group, the host receives *DS0 Status Change* messages of INS.

> **Important!** The circuits that purged because of span failure do not send Circuit Resets in this case because they are included in the Circuit Group Unblocking from the span alarm.

All CICs related to the DPC in question are taken OOS and no blocking messages are sent if the MTP declares one of the following:

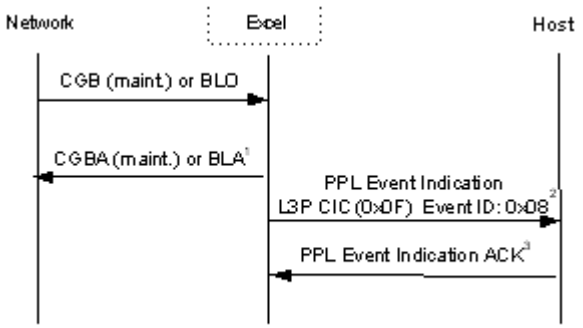
- DPC Inaccessible
- Remote ISUP Unavailable
- MTP Pause

In either case, a General alarm of SS7 Remote ISUP Unavailable (0x0F) is sent to the host.

Circuit Maintenance and Management Call Flows

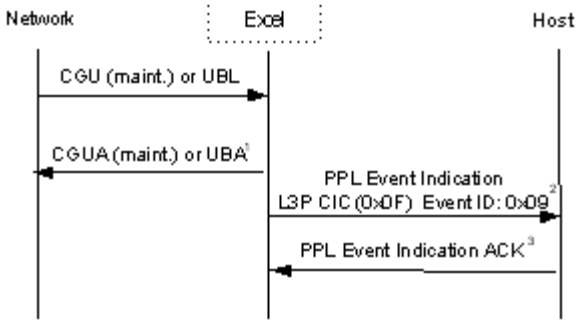
Some messages in these call flows are numbered. The numbered messages are described in more detail beside the corresponding number following each call flow.

Incoming Maintenance Blocking



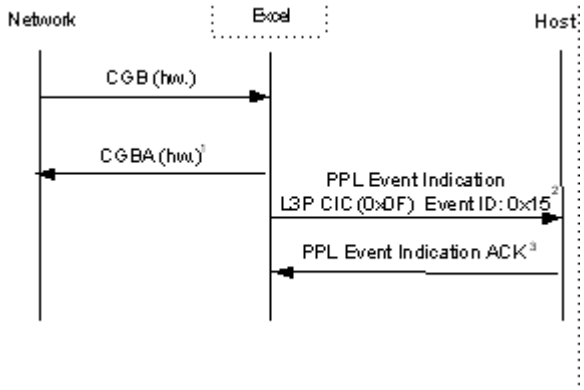
1. The CSP automatically sends the DPC the proper CGBA or BLA.
2. The CSP alerts the host when an incoming Maintenance Blocking Request is made by sending a *PPL Event Indication* message for every channel blocked by the DPC. The *PPL Event Indication* has an SS7 parameters ICB that contains the data of either the incoming CGB or BLO.
3. Upon receiving this message, the host should not send outgoing calls on this channel until the blocking is cleared by the DPC. The DPC may send an incoming call on the blocked channel, however. In this event, the host receives a *PPL Event Indication* of Maintenance Unblocking (without an SS7 parameters ICB) followed by a *Request for Service* message for the incoming call. The host must acknowledge the PPL Event Indication message.
4. The host must acknowledge the *PPL Event Indication* message.

Incoming Maintenance Unblocking



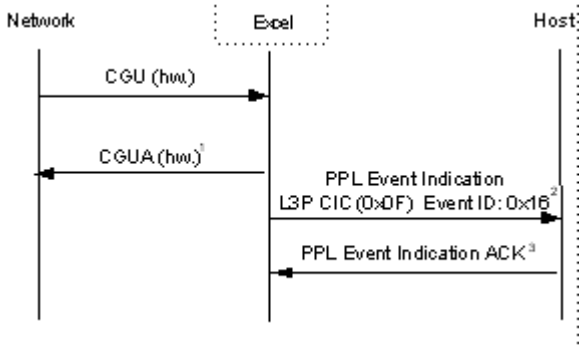
1. The CSP automatically sends the proper CGUA or UBL to the DPC.
2. The CSP alerts the host when an incoming Maintenance Unblocking Request with a *PPL Event Indication* message for every channel unblocked by the DPC. The *PPL Event Indication* has an SS7 parameters ICB containing the data of either the incoming CGU or UBL.
3. Upon receiving this message, the host can make the channel available for outgoing calls. There may be instances in which no SS7 Parameter ICB is included in the *PPL Event Indication* message.
4. The host must acknowledge the *PPL Event Indication*.

Incoming Hardware Failure Oriented Blocking (ITU only)



1. The CSP automatically sends the DPC the proper CGBA.
2. The CSP alerts the host when an incoming Hardware Failure Oriented Blocking Request with a *PPL Event Indication* message for every channel blocked by the DPC. The *PPL Event Indication* message includes an SS7 parameters ICB containing the data of the incoming CGB.
3. Upon receiving this message, the host should not send outgoing calls on these channels until the blocking is cleared by the DPC. The DPC may send an incoming call on the blocked channels, however, upon which, the host will receive a Maintenance Unblocking *PPL Event Indication* (without an SS7 parameters ICB) followed by a *Request for Service* message for the incoming call.
4. The host must acknowledge the *PPL Event Indication* message.

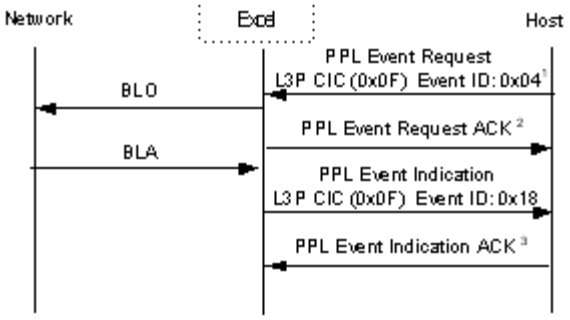
Incoming Hardware Failure Oriented Unblocking (ITU only)



1. The CSP automatically sends the DPC the proper CGUA.
2. The CSP alerts the host when an incoming Hardware Failure Oriented Unblocking Request is made with a *PPL Event Indication* message for every channel blocked by the DPC. The *PPL Event Indication* will include an SS7 parameters ICB containing the data of the incoming CGU.
3. Upon receiving this, the host can now make the channels available for outgoing calls. There may be instances in which no SS7 Parameter ICB is attached to the *PPL Event Indication*. See *Incoming Hardware Failure Oriented Blocking (ITU only) (4-68)*.
4. The host must acknowledge the *PPL Event Indication* message.

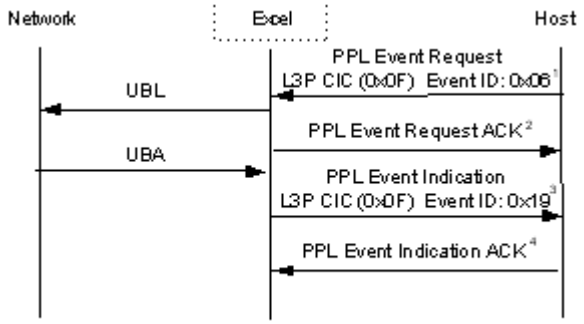
Outgoing Maintenance Blocking Request

By default, the host cannot generate Hardware Failure Oriented Blocking with the *PPL Event Request* message. Hardware Failure Blocking is automatically generated in the event of a loss of signal on the span that carries the CIC. Unblocking is automatic when spans align.



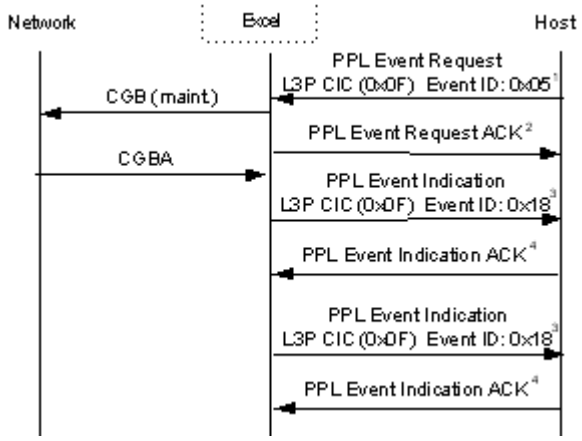
1. The host sends a *PPL Event Request* message to request Local Maintenance Blocking for a CIC. The message can contain an SS7 parameter ICB with parameters for the outgoing BLO message.
2. The CSP will acknowledge the *PPL Event Request*.
3. Upon successful completion of the maintenance blocking procedure, the CSP alerts the host by sending a *PPL Event Indication* message. The *PPL Event Indication* will include an SS7 Parameters ICB containing parameters from the BLA. The host should not initiate any outgoing call without first sending a *PPL Event Request* message of Maintenance Unblock.
4. The host must ACK the *PPL Event Indication*.

Outgoing Maintenance Unblocking Request



1. The host sends a *PPL Event Request* message to request Local Maintenance Unblocking for a CIC. The event request can contain an SS7 Parameter ICB with parameters for the outgoing UBL message.
2. The CSP acknowledges the *PPL Event Request*.
3. Upon successful completion of the Maintenance Unblocking procedure, the CSP alerts the host by sending a *PPL Event Indication* message, which contains an SS7 Parameters ICB containing the parameters from the UBA. The host can now initiate outgoing calls.
4. The host must ACK the *PPL Event Indication* message.

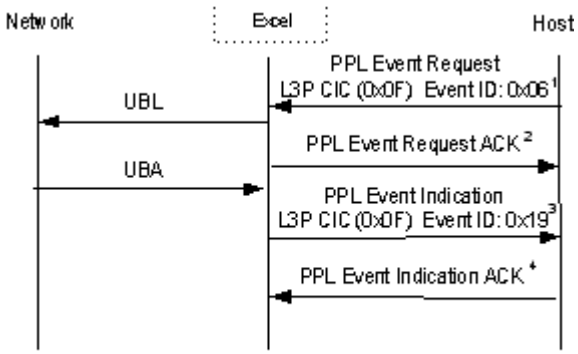
Outgoing Maintenance Group Blocking Request



1. The host sends a *PPL Event Request* message to request Local Maintenance Group Blocking for a CIC circuit group. The event request can contain an SS7 Parameter ICB with parameters for the outgoing CGB message.
2. The CSP acknowledges the *PPL Event Request* message.
3. Upon successful completion of the Maintenance Group Blocking procedure, the CSP alerts the host by sending a *PPL Event Indication* message for each CIC. The *PPL Event Indication* contains an SS7 Parameters ICB, which contains the parameters from the CGBA. The host should not initiate any non-test outgoing call without first sending a Maintenance Unblock *PPL Event Request* message for the channel(s).
4. The host must ACK each *PPL Event Indication* message.

Important! By default, the host cannot generate hardware failure oriented blocking with the *PPL Event Request* message. Hardware Failure Oriented Blocking is automatically generated in the event of a loss of signal on the span that carries the CIC. Unblocking is automatic when the span aligns.

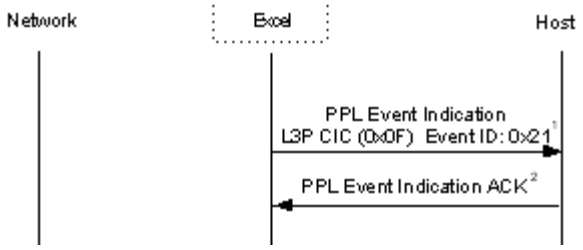
Outgoing Maintenance Group Unblocking Request



1. The host sends a *PPL Event Request* message to request Local Maintenance Group Unblocking for a CIC circuit group. The event request can contain an SS7 Parameter ICB with parameters for the outgoing CGU message.
2. The CSP acknowledges the *PPL Event Request* message.

3. Upon successful completion of the Maintenance Group Unblocking procedure, the CSP alerts the host by sending a *PPL Event Indication* message for each CIC. The *PPL Event Indication* contains an SS7 Parameters ICB containing the parameters from the CGU. The host can now initiate outgoing calls.
4. The host must ACK each *PPL Event Indication* message.

Local Circuit Maintenance Unblocking Indication



1. The CSP alerts the host when the Local Maintenance Blocking state is cleared. This is typically done when an ISUP component has requested that the CIC be reset (for example, receipt of Unreasonable Signaling to an idle ISUP CPC). The host now considers the channel as not Locally Maintenance Blocked. The host can re-establish blocking. See “Outgoing Maintenance Group Blocking Request.”
2. The host must ACK each *PPL Event Indication* message.

CRM/CRA (ANSI)

Definitions

The messages related to circuit reservation are defined in this section.

CRM

Circuit Reservation Message: This message is used to perform continuity procedures on an outgoing SS7 circuit before collecting address digits from the incoming MF circuit. On receiving a seize from the incoming MF agency, a CRM with the required NOC is sent to the succeeding exchange and the timer TCRM (typically 15 seconds) is started. Based on the NOC, a continuity check may or may not be performed.

CRA

Circuit Reservation Acknowledgment: On receipt of a CRA message, the timer TCRM is stopped and further processing continues. If TCRM expires before the receipt of CRA, the CRM is re-attempted on a different CIC.

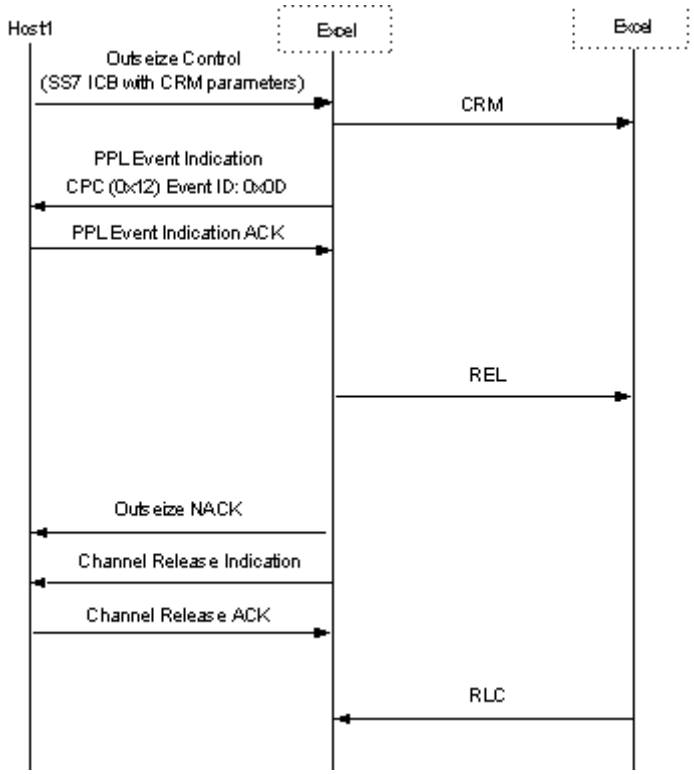
Circuit Reservation Timer

After sending a CRA in response to the CRM, a Timer TCRA (typically 15 seconds) is started by the terminating node while waiting for an IAM (or COT if a continuity check is to be performed). If neither

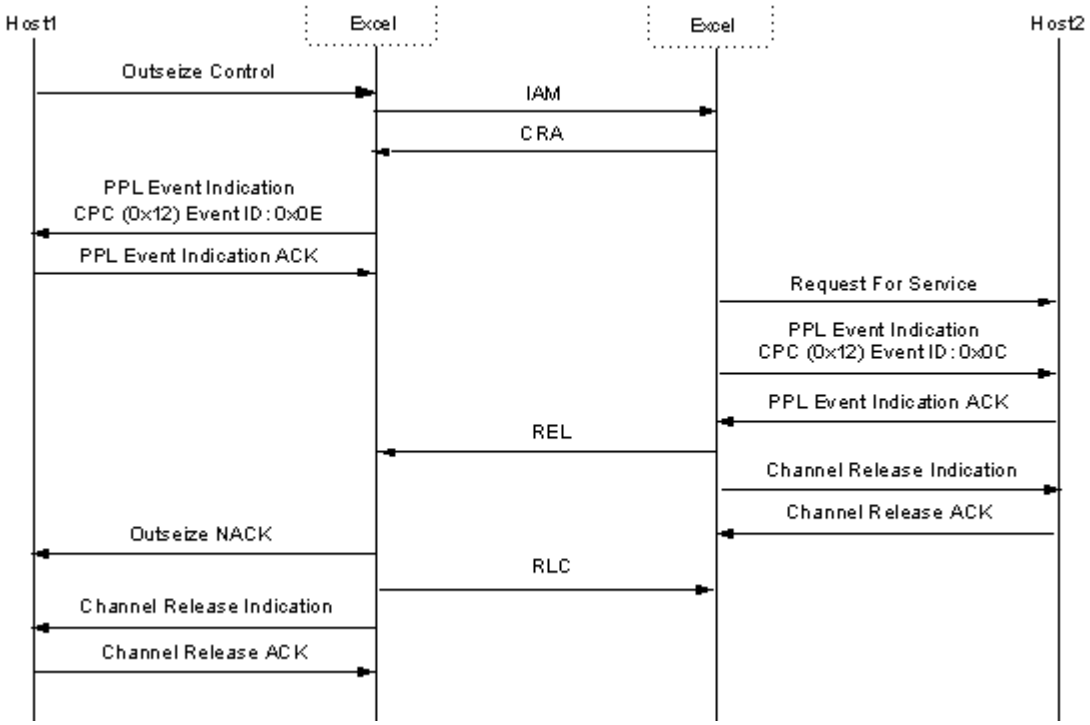
of the messages is received and the timer expires, an REL message is sent from the terminating node with a cause value of “Temporary Call Flows.

Call Flows

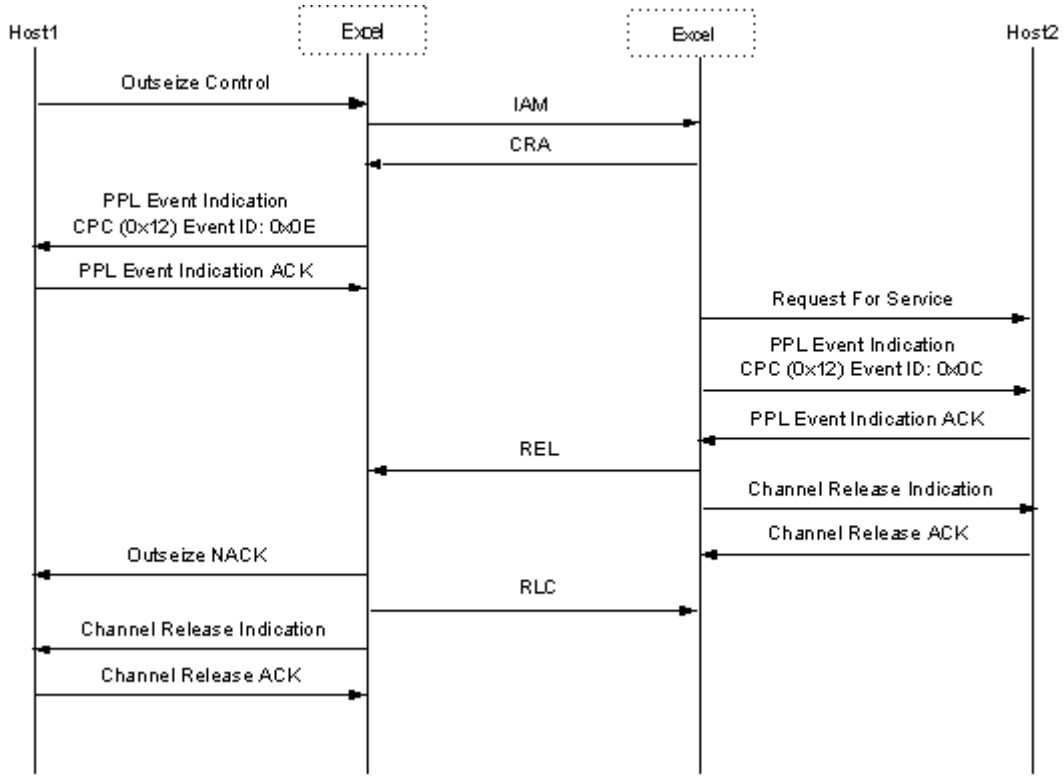
TCRM Timeout



TCRM Expiry



Dual Seizure during CRM Processing



MCP/PCP (ITU)

Overview MCP and PCP messages provide support for unrecognized signaling as defined by ISUP 1992 (ITU-T White Book). This support becomes necessary, for example, if the exchange receives unrecognized signaling information from a later version of ISUP. With MCP/PCP, the exchange can take the appropriate action and ensure its compatibility with new signaling messages, no matter what version they may be.

MCP Message Compatibility Procedure (MCP, Section 2.9.5, Q.764)

It may happen that an exchange receives an unrecognized message. The message compatibility procedures are invoked in this case to ensure predictable network behavior. Message compatibility information is received in the unrecognized message and following actions can be taken based on the compatibility information contained in the message.

Procedures include:

- Send Notification to the exchange sending unrecognized message (Confusion).
- Do not send notification.
- Initiate release procedure.
- Discard message.
- Pass on unrecognized message transparently.

PCP Parameter Compatibility Procedure

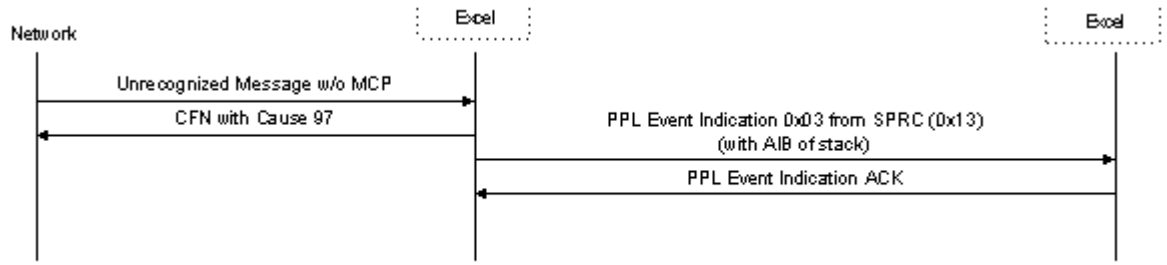
PCP defines a set of procedures executed when a message containing unrecognized optional parameters is received by the exchange. Parameter compatibility information is contained in the received message. Receipt of unrecognized parameters only refers to optional parameters and unexpected parameters for that particular message. Based on the instruction indicator contained in the parameter compatibility information for the unrecognized parameter, the following actions can be taken.

- Send Notification to the exchange sending unrecognized parameter (Confusion).
- Do not send notification.
- Initiate release procedure.
- Discard message.

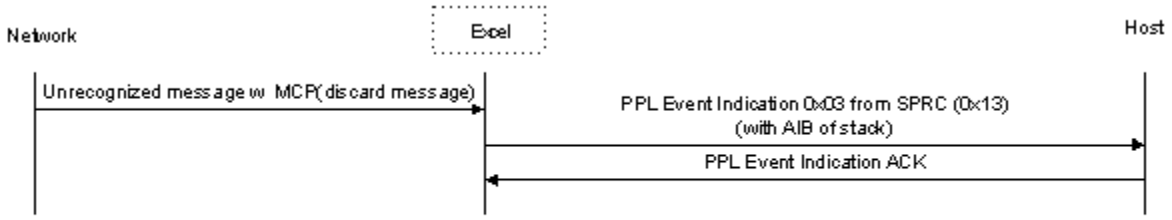
- Discard unrecognized parameter.
- Pass on unrecognized parameters transparently.

Unrecognized Message Handling—for Message without MCP

.

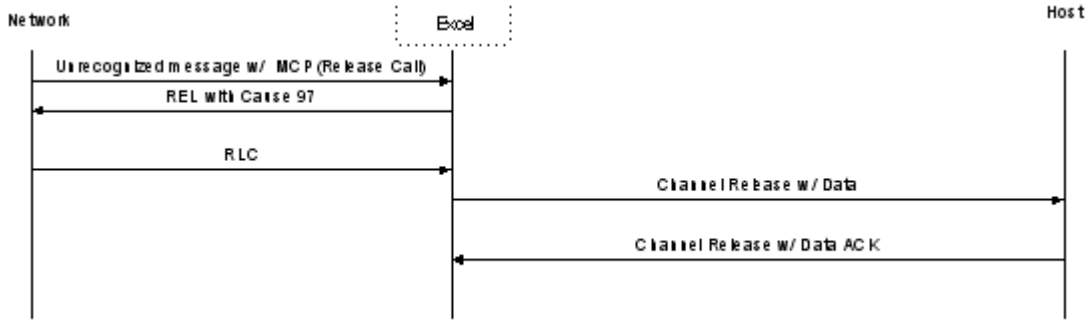


Unrecognized Message Handling---for Message with MCP Discard Indicator

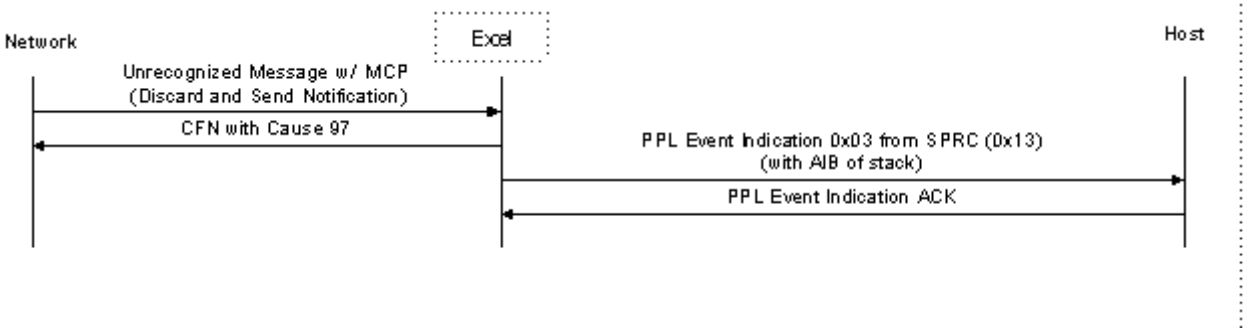


Unrecognized Message—with MCP Containing Release Indicator

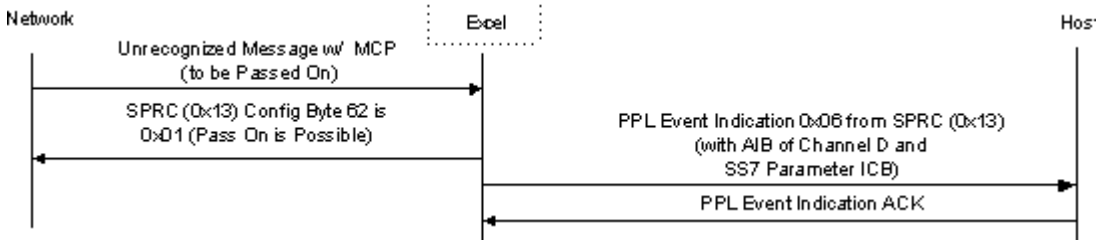
.



Unrecognized Message—with MCP Containing Discard and Send Notification Indicator

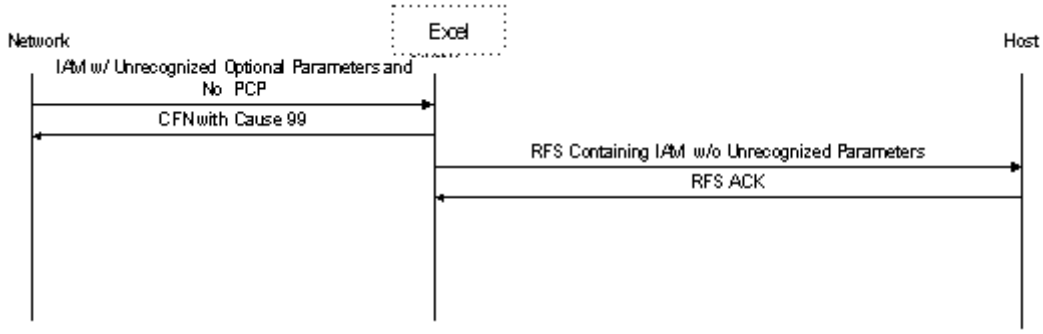


Unrecognized Message--with MCP Containing Pass On Indicator

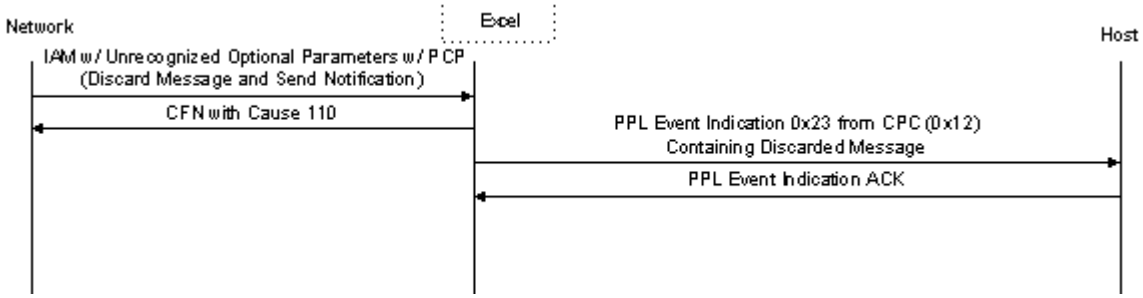


Message with Unrecognized Optional Parameters and no PCP

.

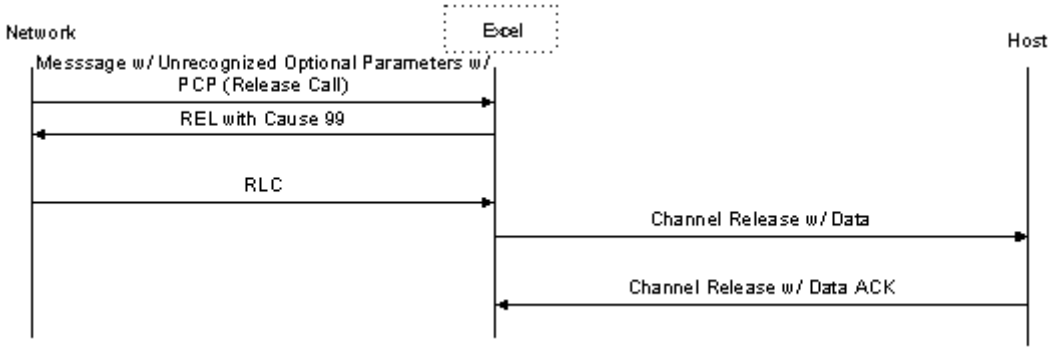


Message with Unrecognized Parameter—with PCP Containing Discard Message and Send Notification Indicator for that Parameter

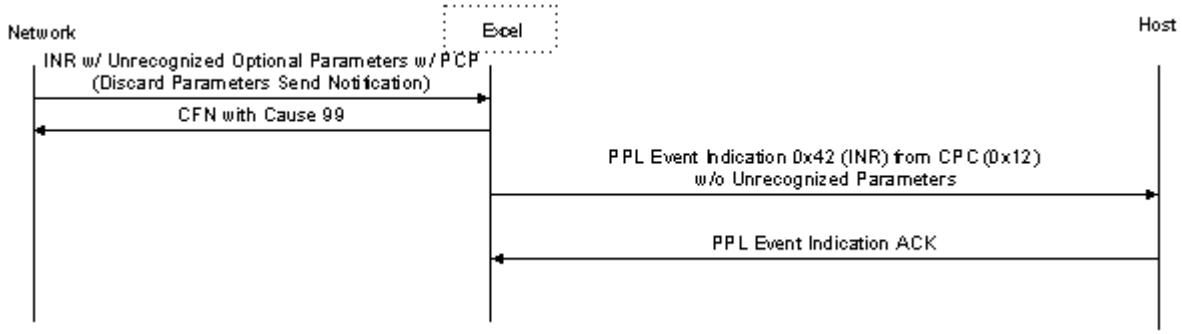


Message with Unrecognized Parameter—with PCP Containing Release Call Indicator

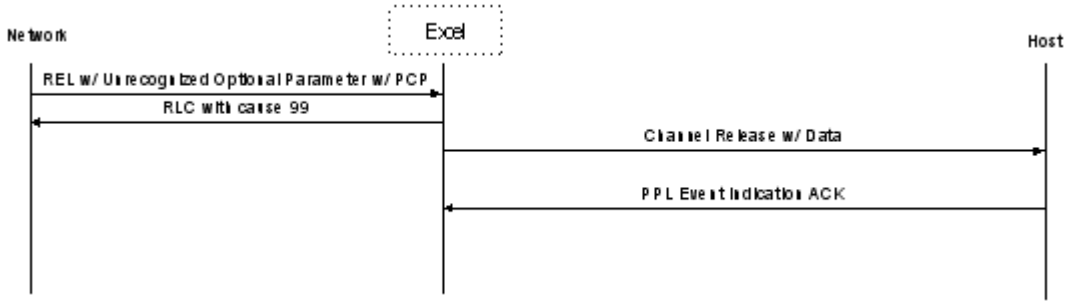
.



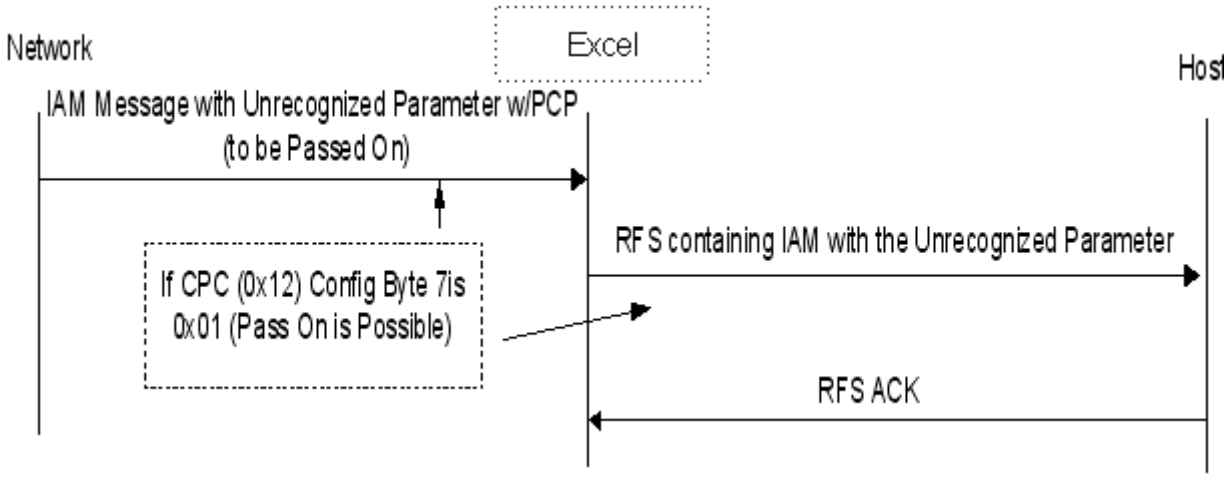
Message with Unrecognized Optional Parameter—with PCP Containing Discard Parameter and Send Notification Indicator



Release with Unrecognized Parameter with PCP



Message with Unrecognized Parameter and PCP Having Pass On Indicator



MPM/CCL/OPR

Overview

This section includes information and call flows related to MPM/CCL/OPR. MPM, CCL and OPR messages are implemented as defined by CHINA ISUP specifications.

Definitions

MPM

Meter Pulse Message: A message is sent in the backward direction after Answer at equal intervals (usually every one minute).

CCL

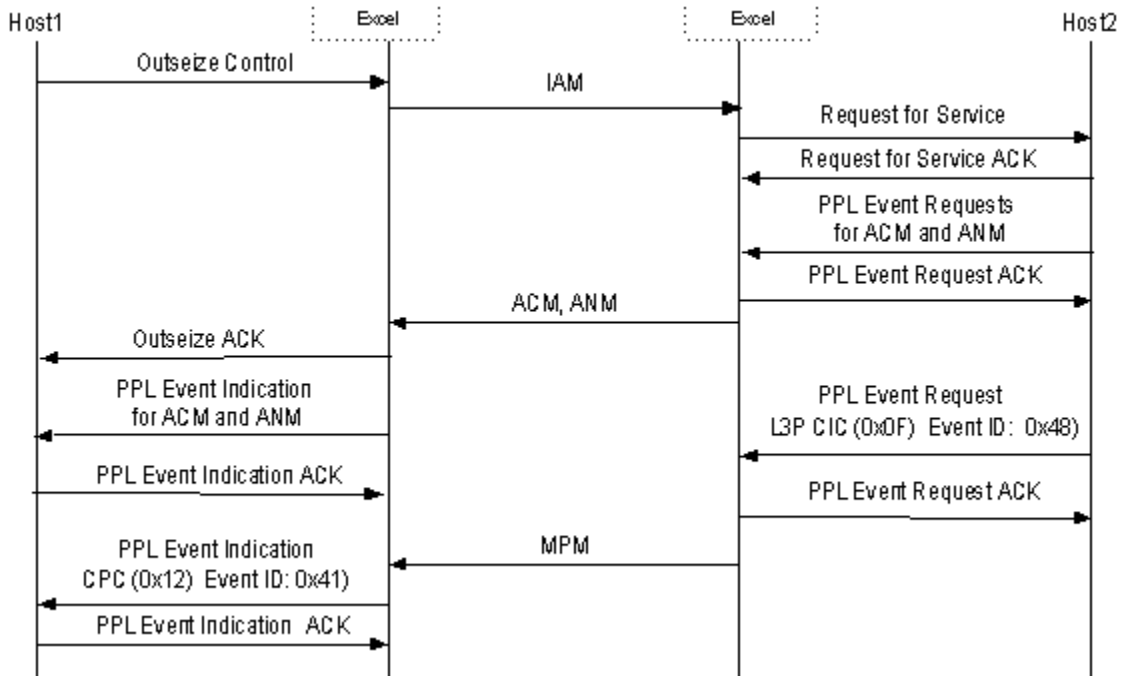
Calling Party Clear Message: This message is sent in the forward direction when the calling party, which has called the special service node, wants to release the call.

OPR

Operator Message: This message is sent in the backward direction from the special service node if the calling party needs to be provided with a ring after hanging up.

Call Flows

MPM - Normal

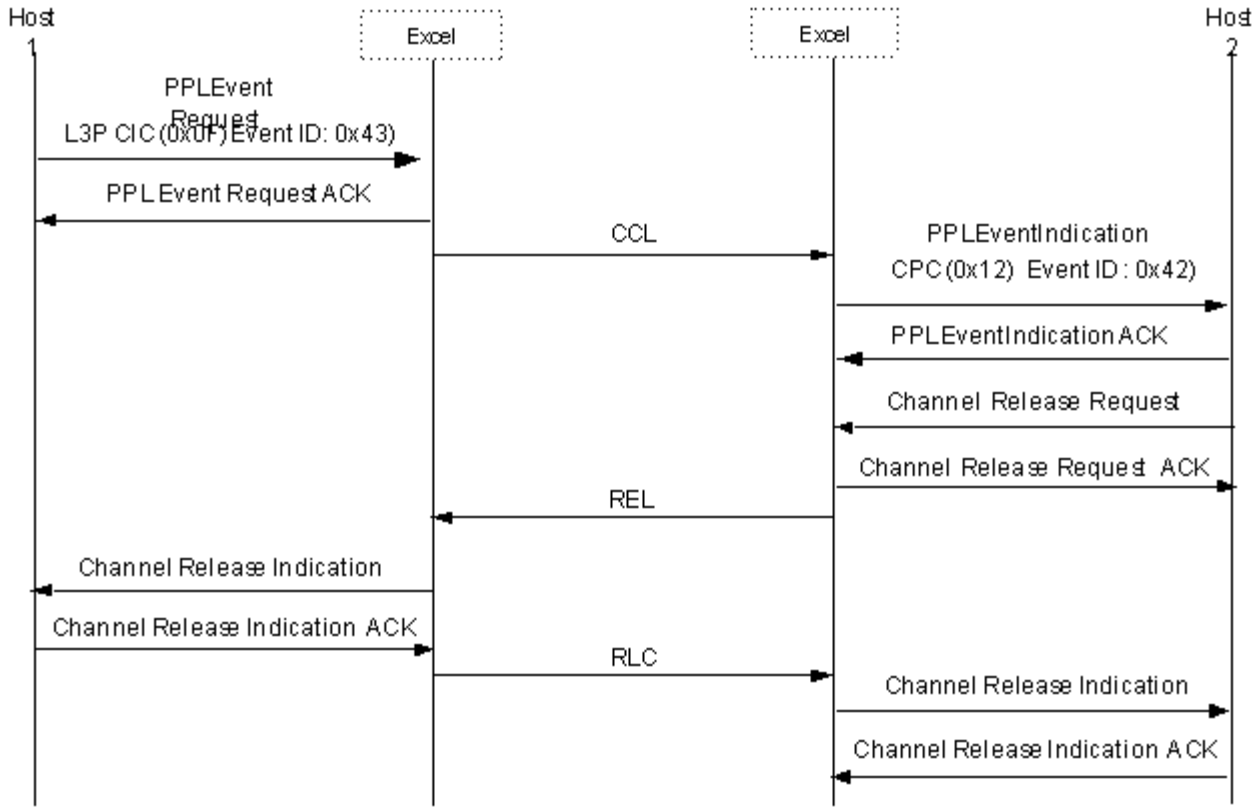


Calling Party Clear—Normal Release Sequence

The normal release sequence for Calling Party Clear (ITU). The call flow starts with the call in the answered state:

Host 1 = Calling Party

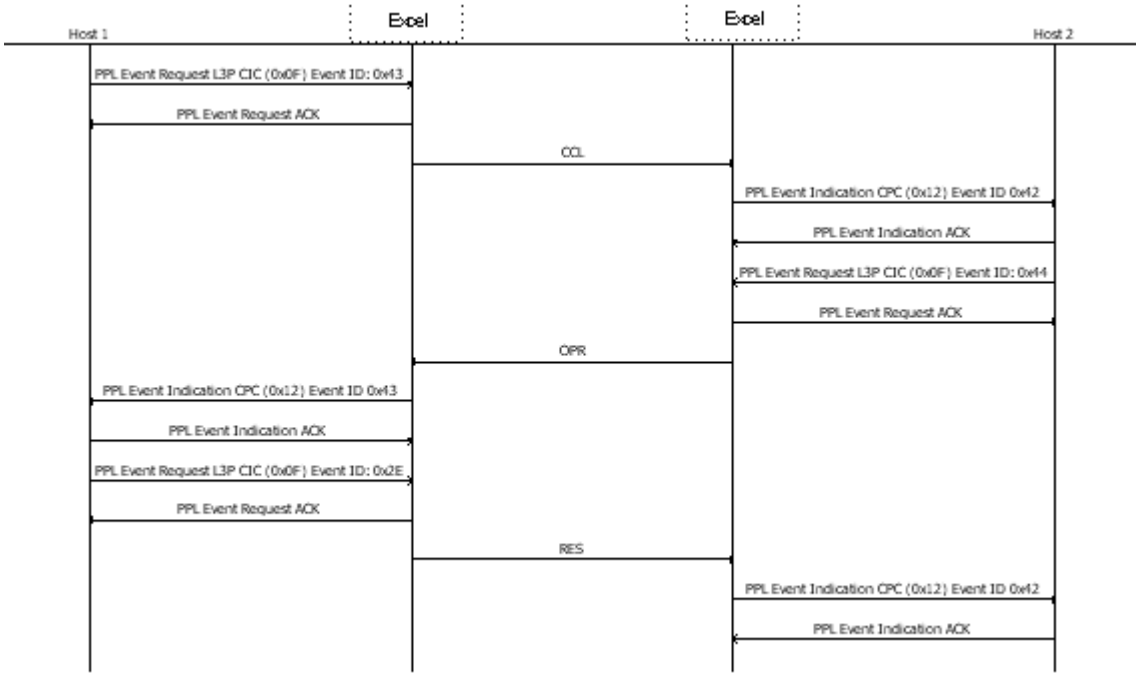
Host 2 = Called Party



Calling Party Clear—Operator Callback—Subscriber Re-answers

Shows the normal release sequence for Calling Party Clear (ITU) with an operator callback. The call flow starts with the call in the answered state:

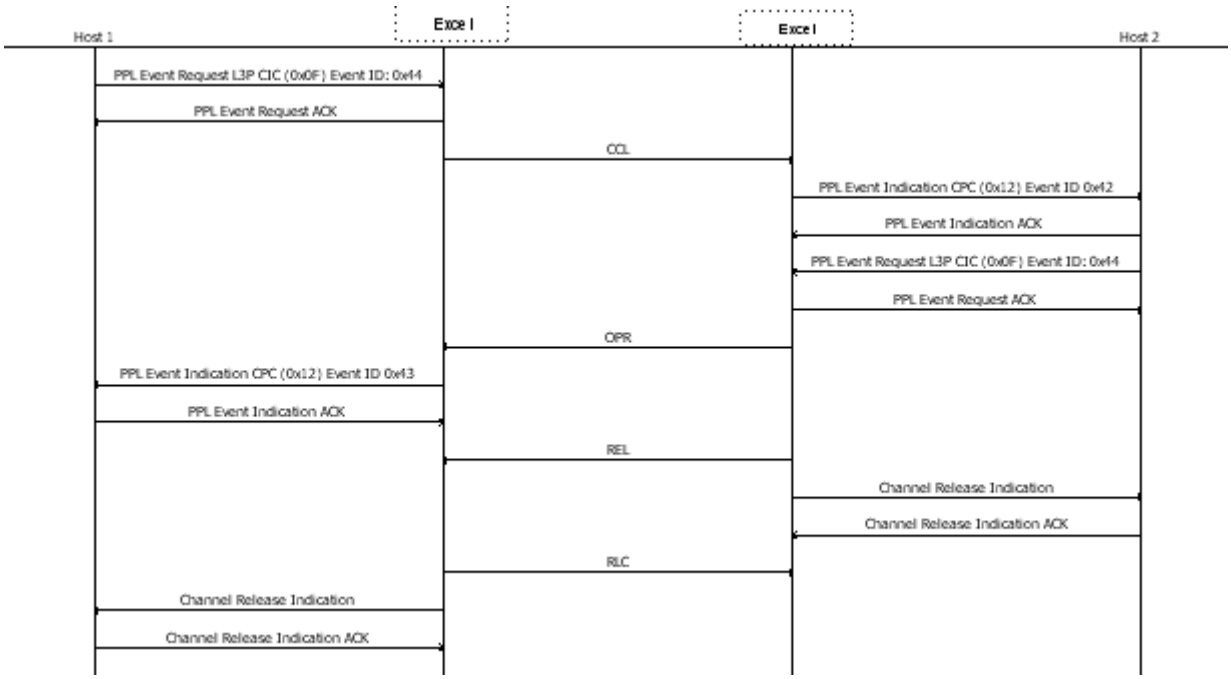
Host 1 = Calling Party
Host 2 = Called Party



Calling Party Clear—Operator Callback—Re-Answer Timeout

This call flow shows the normal release sequence for Calling Party Clear (ITU) with an operator callback and a re-answer timeout. The call flow starts with the call in the answered state:

Host 1 = Calling Party
Host 2 = Called Party



Suspend/Resume

Overview

This section includes information and call flows about the ISUP Suspend and Resume messages.

Definitions

Suspend

The Suspend (SUS) message indicates a temporary cessation of communication without releasing the call. SUS can only be accepted in an answered state (ICC answered/OGC answered).

Resume

A Resume (RES) message indicates a request to recommence communication. If the RES is not sent within timer T6 or T2 (T2 applies to ITU only), the controlling exchange will initiate a release procedure.

PPL Information

The SUS and RES messages are passed in the *PPL Event Request* and *PPL Event Indication* messages, with the following Event ID values:

0x2D	SUS
0x2E	RES

For ANSI, only network-initiated SUS/RES are allowed. Only called party can send SUS. For ITU, both user and network-initiated SUS/RES are supported. Both calling and called party can initiate SUS.

Atomic function (104) in the ISUP CPC component checks for network- or user-initiated SUS/RES.

Host-Initiated

L3P CIC passes the SUS request from the host to CPC, upon which CPC enters the suspended state and waits for a RES request to recommence the communication. L3P CIC forwards the RES request from the host to CPC and CPC returns to the answered state. If the RES is not received within the resume wait timer, the call is released.

Network-Initiated

When a SUS is received from the network, CPC enters the suspended state and waits for a RES. When a RES is received from the network, CPC returns to the answered state. If the timer initiated at the controlling exchange expires before RES is received, REL is initiated.

PPL event indications include data in SS7 Raw ICB format. If the SUS/RES is not received with the valid indicator, a *PPL Event Indication* message indicating the error is sent to the host.

Customization

Use the PPL configuration bytes in the L3P CIC component to modify the handling of the SUS and RES messages.

SUS (ITU)

The table below shows the L3P CIC PPL Configuration Bytes used to configure the SUS message for ITU.

Byte	Description	Value
485	Information Length	0x05
486	Message ID	0x0D
487	Number Of Parameters	0x01
488	Parameter1 ID	0x22 (Suspend/Resume indicator)
489	Parameter Data Length	0x01
490	Parameter value	0x01 (Network Initiated)

SUS (ANSI)

The table below shows the L3P CIC PPL Configuration Bytes used to configure the SUS message for ANSI.

Byte	Description	Value
480	Information Length	0x05
481	Message ID	0x0D
482	Number Of Parameters	0x01
483	Parameter1 ID	0x22 (Suspend/Resume indicator)

Byte	Description	Value
484	Parameter Data Length	0x01
485	Parameter value	0x01 (Network Initiated)

RES (ITU)

The table below shows the L3P CIC PPL Configuration Bytes used to configure the RES message for ITU.

Byte	Description	Value
492	Information Length	0x05
493	Message ID	0x0E
494	Number Of Parameters	0x01
495	Parameter1 ID	0x22 (Suspend/Resume indicator)
496	Parameter Data Length	0x01
497	Parameter value	0x01 (Network Initiated)

RES (ANSI)

The table below shows the L3P CIC PPL Configuration Bytes used to configure the RES message for ANSI.

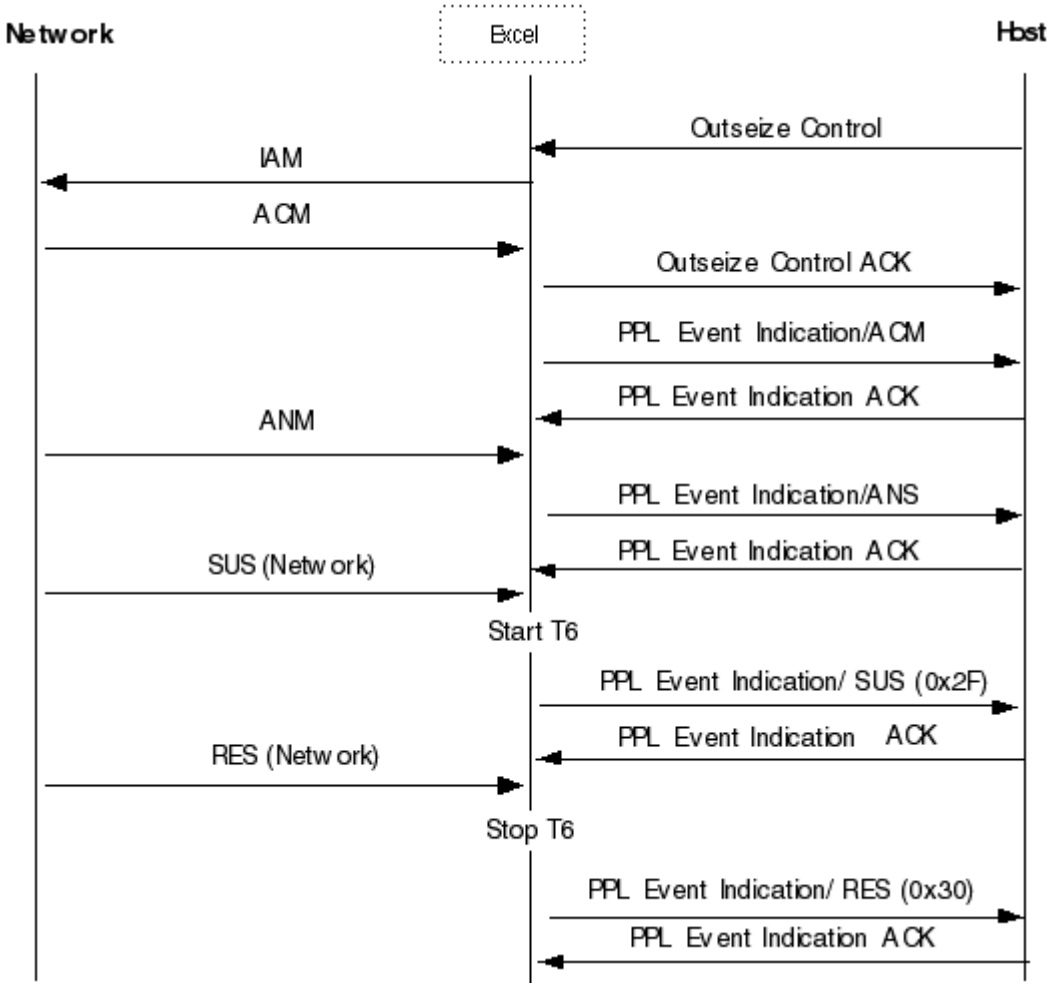
BYTE	Description	Value
490	Information Length	0x05
491	Message ID	0x0E
492	Number Of Parameters	0x01
493	Parameter1 ID	0x22 (Suspend/Resume indicator)
494	Parameter Data Length	0x01
495	Parameter value	0x01 (Network Initiated)

Suspend and Resume Call Flows

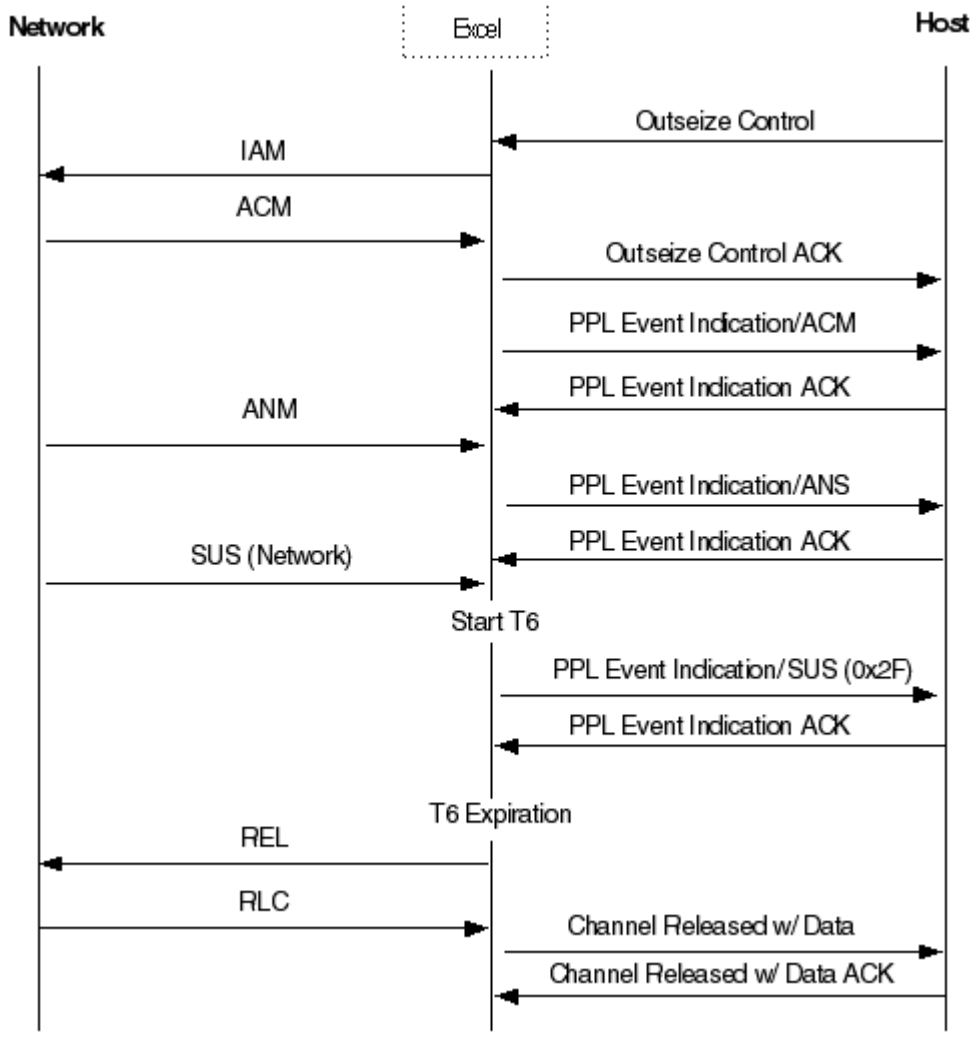
This section includes call flows showing the Suspend and Resume messages

Suspend Followed by Resume

.



Suspend Followed by Release



UPT/UPA (ITU)

Overview This section defines UPT/UPA and includes call flows related to these messages.

Definitions

UPT

User Part Test Message: The User Part Test (UPT) message is used to test whether a user part is available on a signaling point, which has been marked inaccessible. The test procedure is initiated when the ISUP receives an MTP STATUS Primitive with cause – “user part unavailability – inaccessible remote user”. The ISDN User Part shall send a User Part Test message and start timer that supervises the receipt of a response to the user part test message. All MTP STATUS Primitives with cause – “user part unavailability – inaccessible remote user” are ignored when the timer T4 is running. The timer T4 is stopped when a User Part Available (UPA) or any other ISUP message is received.

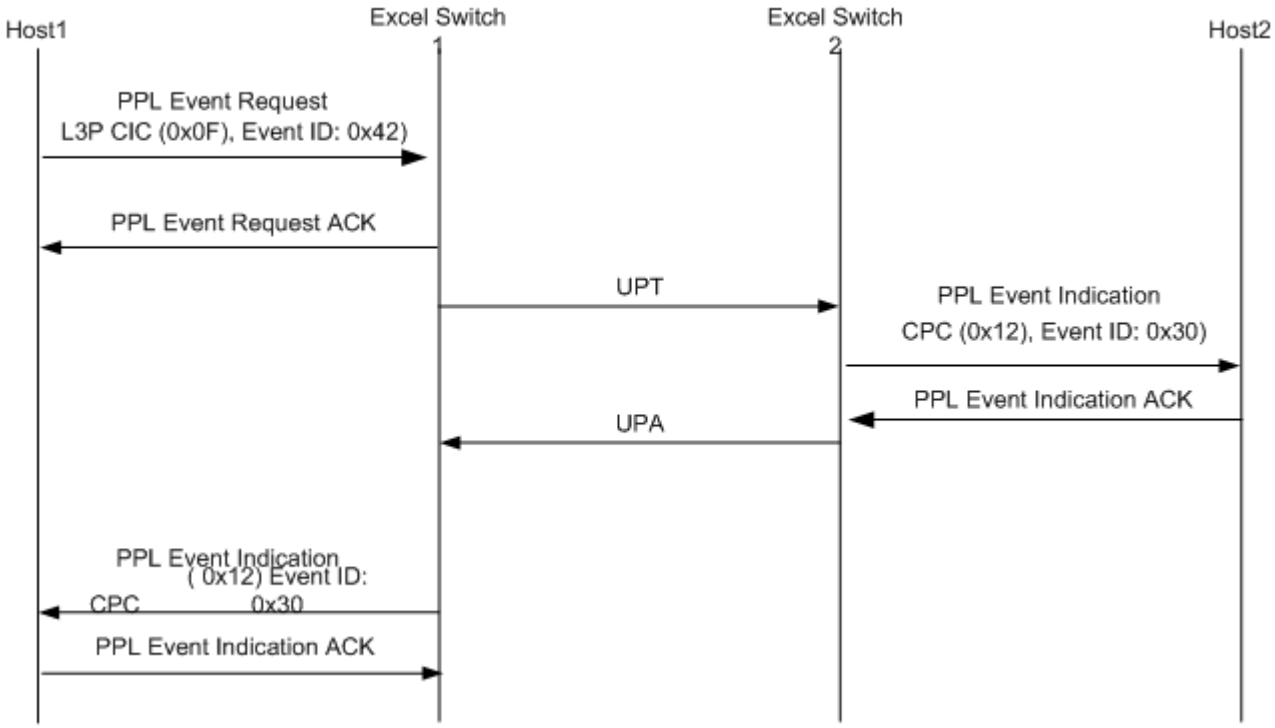
UPA

User Part Available: When a User Part Test acknowledgment message is received by ISUP, the ISUP sends a User Part Available message in response to the message.

UPT and UPA messages are implemented for ITU/CHINA ISUP, with procedures as recommended by ITU ISUP (Q.764).

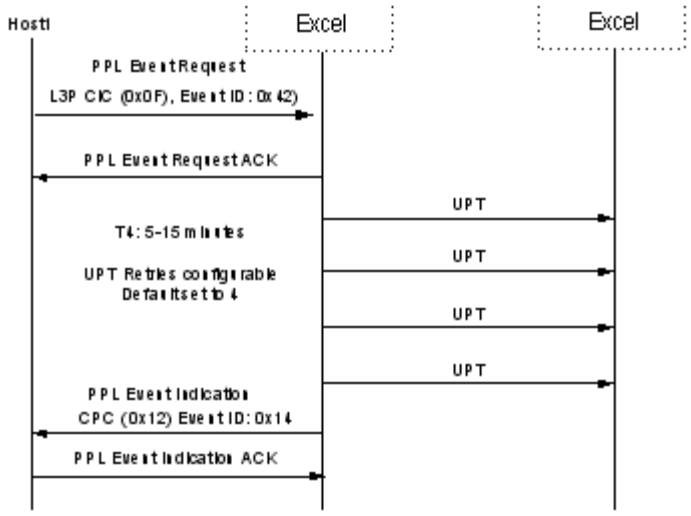
Call Flows Normal Call Flow for UPT/UPA

.



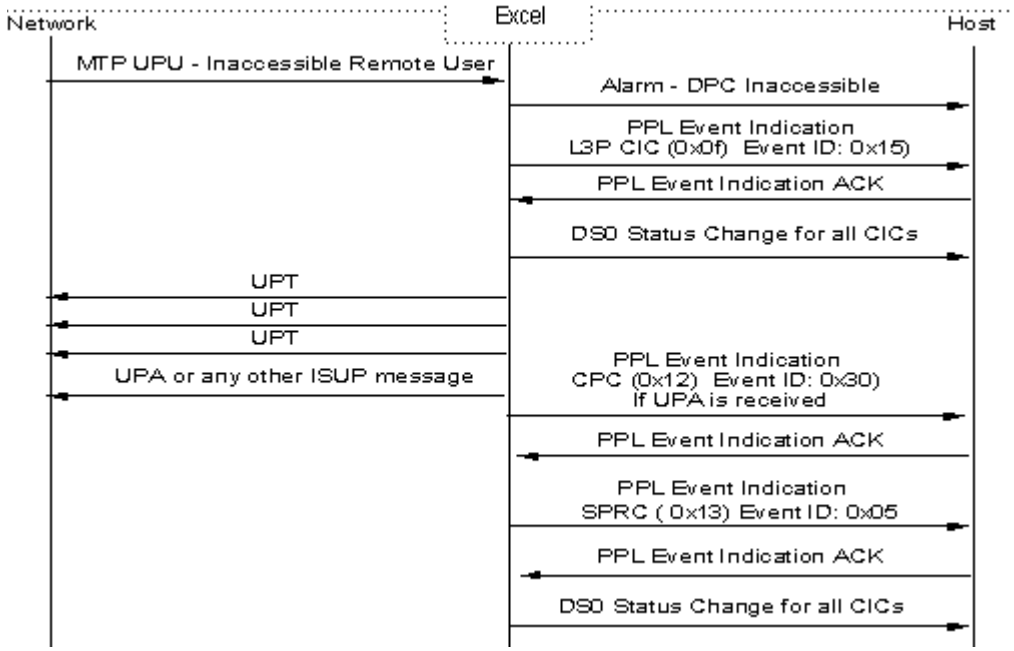
Call Flow for UPA Timeout

.



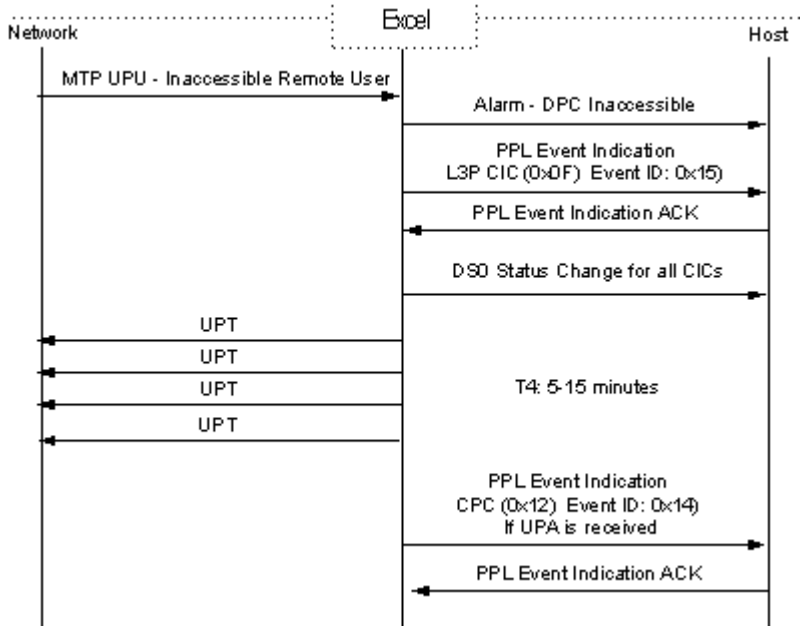
UPT Procedure: Remote User Unavailable

The call flow shows the UPT procedure that is initiated when ‘MTP-UPU — Remote user unavailable’ is received.



UPT Procedure—No Response from Far End

This shows the UPT procedure that is initiated when ‘MTP-UPU – Remote user unavailable’ received, and there is no response from the far end.



SS7 Customization with ISUP Messages

Overview

You can customize the following to meet your application requirements:

- ISUP Message Configuration
- User-defined ISUP Messages
- ISUP MTP Pause Logic Options
- *Request For Service* Message Format
- SS7 Parameter Presentation
- PPL Configuration Bytes
- Protocol Timers
- Other Customization

ISUP Message Configuration

The ISUP Message Configuration Template is a table against which outgoing ISUP messages are formatted and incoming message parameters are extracted and verified. The default configurations for ANSI are described in the ANSI T1.113 specification, and the default configurations for ITU are described in ITU Q.767 specification.

The *SS7 ISUP Message Format Configure* message allows the host to customize the following ISUP message contents:

- Message type
- Priority
- Number, order, and length of all parameters

You can also customize messages for ISUP variants. There is one ISUP Message Configuration Template for each protocol stack. All SS7 channels assigned to a stack are affected by modifications to the corresponding table. See the *SS7 ISUP Message Format Configure* message for an example, as well as generic sections on downloading and assigning modified PPL components and their relationship to stack configuration.

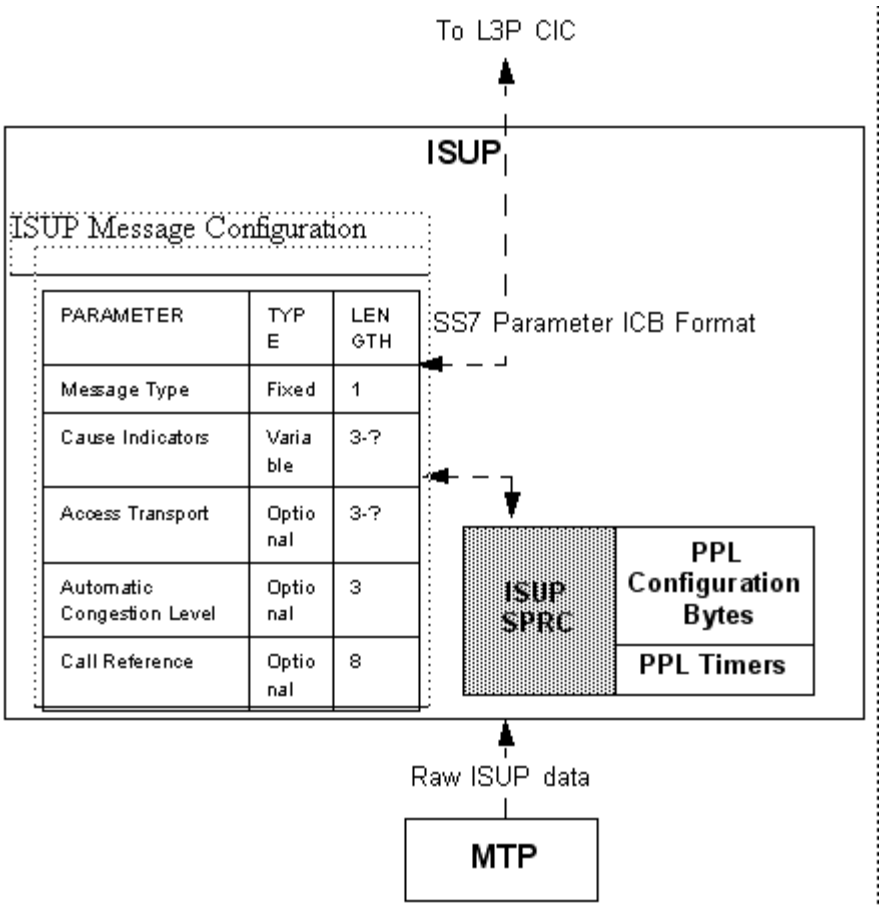
As shown in the figure below, the raw ISUP data from MTP is compared against the template to identify the ISUP message type and convert the raw data into the format of an SS7 Parameters ICB to send to L3P.

Incoming and outgoing messages that do not match the template, cause the host to be notified of a protocol violation and the CSP discards the message.

For outgoing calls, the parameter data is converted from an SS7 Parameters ICB format into the raw data format to go out to the network.

You can modify the defaults to include (or to exclude) optional parameters and customize messages for ISUP variants. Changing the format of a message affects the formatting of every incoming and outgoing instance of that message.

Figure 4-2 ISUP Message Configuration Template

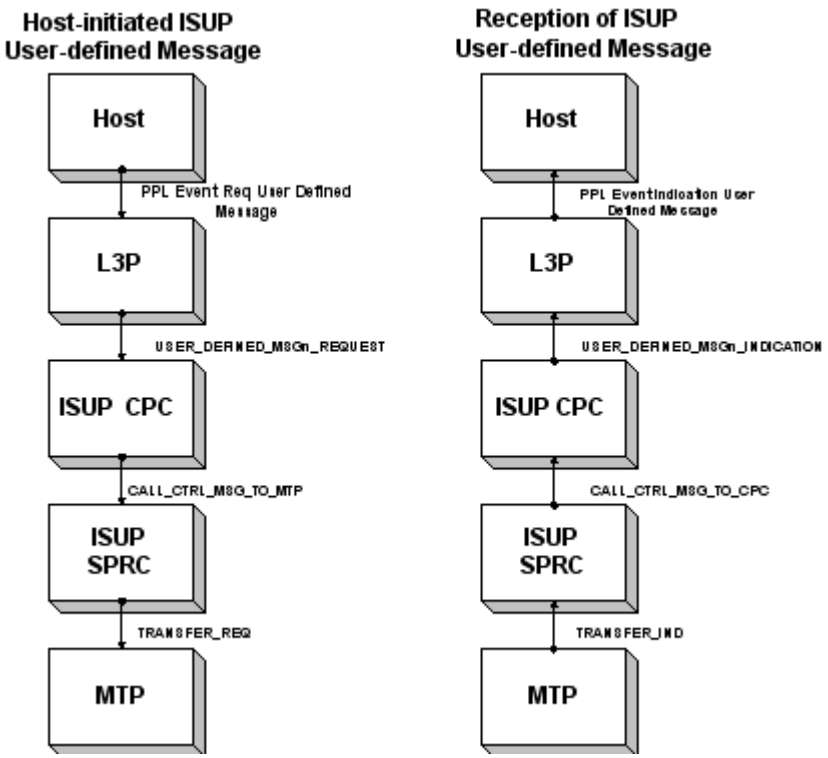


User-defined ISUP Messages

SS7 allows the host to configure up to 32 user-defined ISUP messages per stack. The host can create PPLs to send and receive user-defined ISUP messages.

The L3P CIC and ISUP CPC components must be modified to create the Transport Event User-defined Messages. The *Figure 4-3, User-defined ISUP Message Mode (4-97)* illustrates the model for sending and receiving user-defined ISUP messages.

Figure 4-3 User-defined ISUP Message Mode



To facilitate the passing of user-defined ISUP messages, add events and atomic functions to ISUP CPC and L3P CIC. L3P CIC Atomic Function (AF) 99 allows L3P to send a user-defined ISUP message to ISUP. ISUP CPC AF 64 sends an incoming user-defined ISUP message to L3P. AF 65 sends an outgoing user-defined ISUP message to SPRC. For an example of L3P CIC customization, see *Figure 4-4, L3P CIC Send INF/Receive INR (4-100)* and *Figure 4-5, L3P CIC Send INR/ Receive INF (4-101)*.

Example Using User-defined ISUP Message Procedure

The following example shows how to use the User-defined ISUP message procedure.

1. Use the PPL Event Request / Indication messages to send or to receive User-defined Message (UDM).

   For event values of UDMs, use any value larger than 30 (0x1E). For example, use the following to handle the receiving of INR / INF and the sending of INF/INR:

   - 32(0x20) INR Indicator Event ID
   - 33 INF Indicator Event ID

   Then, they are used for L3PCIC Host-side Event Request ID:

   - INR (532)PPLevL5_EVENT_REQ_32
   - INF (533)PPLev:5_EVENT_REQ_33

   For the L3PCIC network side:

   - INR (81) L3PCICevISUP_User_Defined_Msg1_IND
   - INF (82) L3PCICevISUP_User_Defined_Msg2_IND

   For ISUP CPC, you do not need to assign an event ID. When you define the UDM in the ISUP Message Format Configuration template (MCT), the PPL automatically assigns the event ID. For example, you define:

   - INR User-defined message #10x2F(47) (ISUP MCT index)
   - INF User-defined message #20x30(48) (ISUP MCT index)

   Then, the ISUP CPC creates:

   - Events (101)CPCevSPRC_USER_DEFINED_MSG1 —>INR
   - (102)CPCevSPRC_USER_DEFINED_MSG2 —>INF
   - (151)CPCevL3P_USER_DEFINED_MSG1_REQ —> Outgoing INR
   - (152)CPCevL3P_USER_DEFINED_MSG2 —->Outgoing INF

2. Modify ISUP Message Configuration Table (MCT).

   To modify the ISUP Message Configuration Table, create the message templates for the new messages according to the related specification. Use the *SS7 ISUP Message Format Configure* API message (0x6B) to assign the UDM number. For example:

   - INR —>2F(47)
   - INF —>30(48)

3. Modify the PPL State Machine.

   To modify the PPL State Machine, decide which state to modify or add. Decide which state will receive or send the UDMs. For example, you will receive an INR after an IAM, which means the L3P CIC is in the Inseize state (State 2).

   For L3P CIC - You can see the following DS0 file for example, but you have to pay attention to sending UMD and receiving UDM.

   AF 76 adds prestored ISUP parameter entities into the L3P CIC configuration bytes. The following examples uses configuration bytes 150-152 and 160-162. If no parameters are prestored, use 0 as the number of parameters.

150	Information Length	0x02 (INR)
151	Message ID	0x03
152	Number of Parameters	0x00 (no per-stored)
160	Information Length	0x02 (INF)
161	Message Id	0x04
162	Number of Parameters	0x00 (no per-stored)

   Next, fill the AF76(160) for INF message.

   - AF76(150) for INR in order to use these PPL configurative byte entities
   - AF71 Fill in the ISUP MSG Index, for example AF71(48), where 48 is the ISUP MSCT index 48(0x30) is the MCT.

   For receiving UDM, the event IDs include the following:

   - (101)L3PCICevISUP_User_defined_MSG1_IND —->INR
   - (102)L3PCICevISUP_User_Defined_MSF2_IND —->INF

   For CPC - For the EVENT ID and ISUP MCT index, refer to the DSD file.

4. Create PPL report file and download to the CSP.

   Use the PPL tool to create state table function to generate the PPL REP file. Then, use the tool REP2CFG.EXE to create the CFG files.

   Finally, define the user protocol ID and download the report to the CSP.

Figure 4-4 L3P CIC Send INF/Receive INR

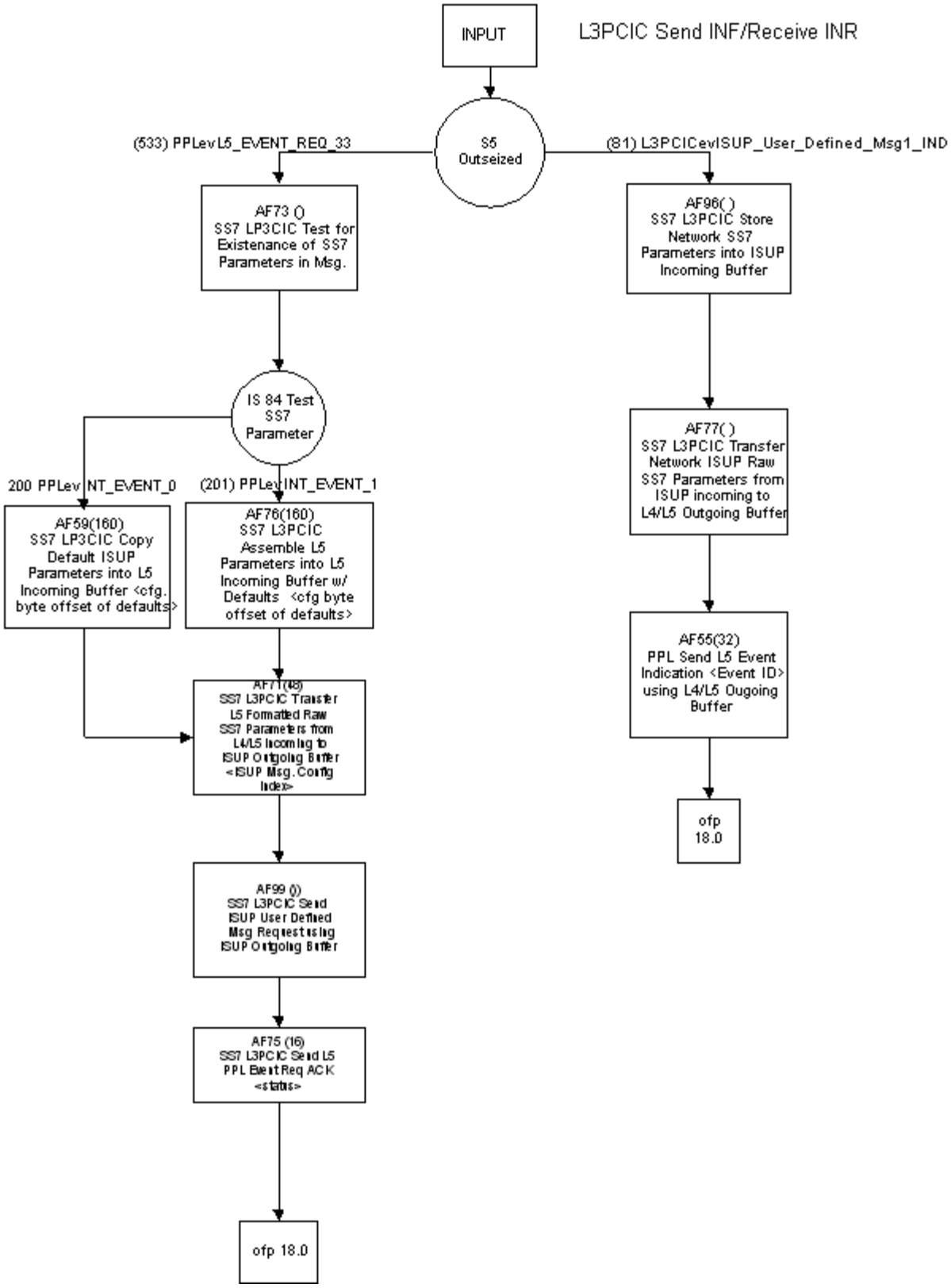


Figure 4-5 L3P CIC Send INR/Receive INF

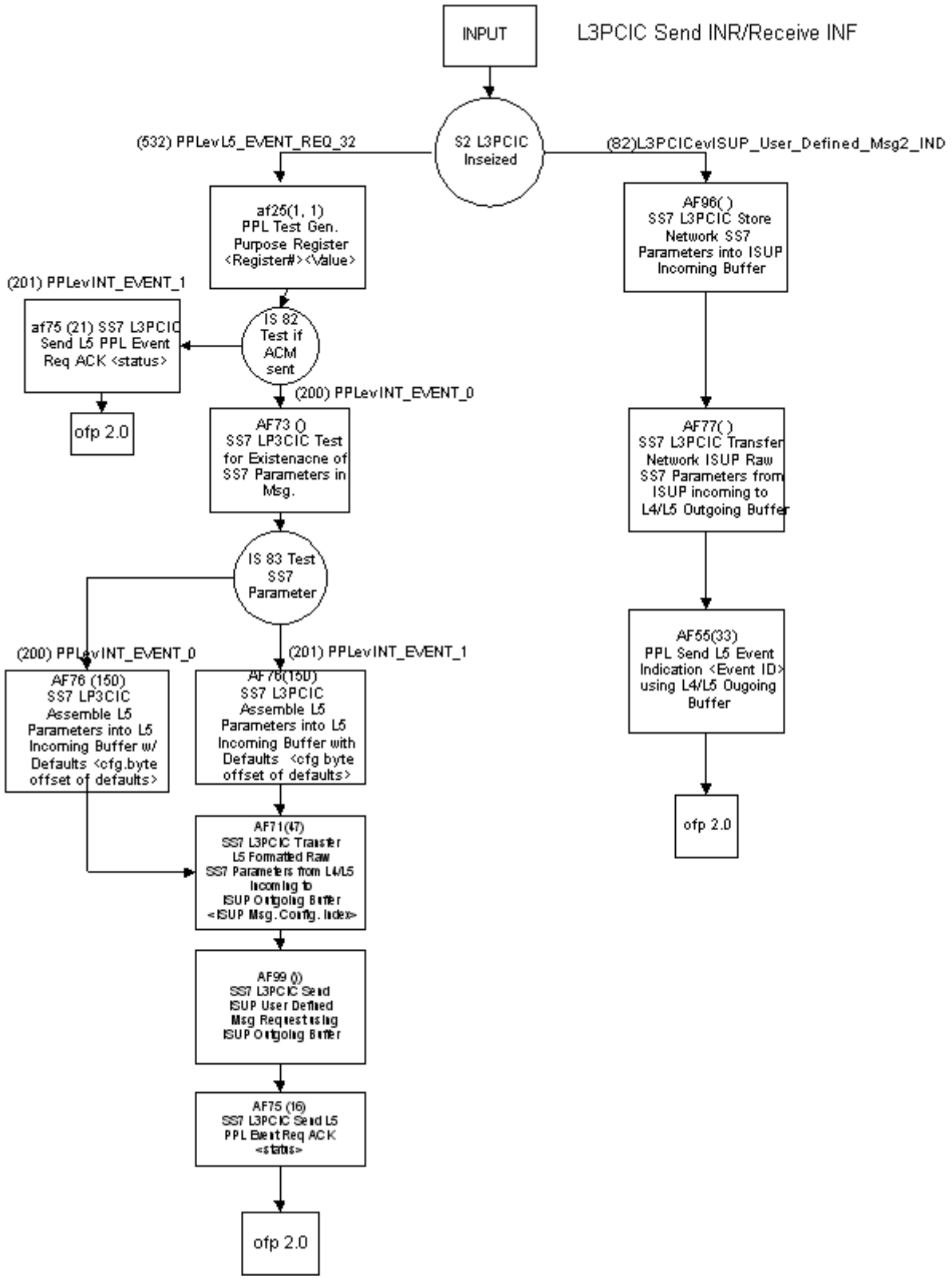
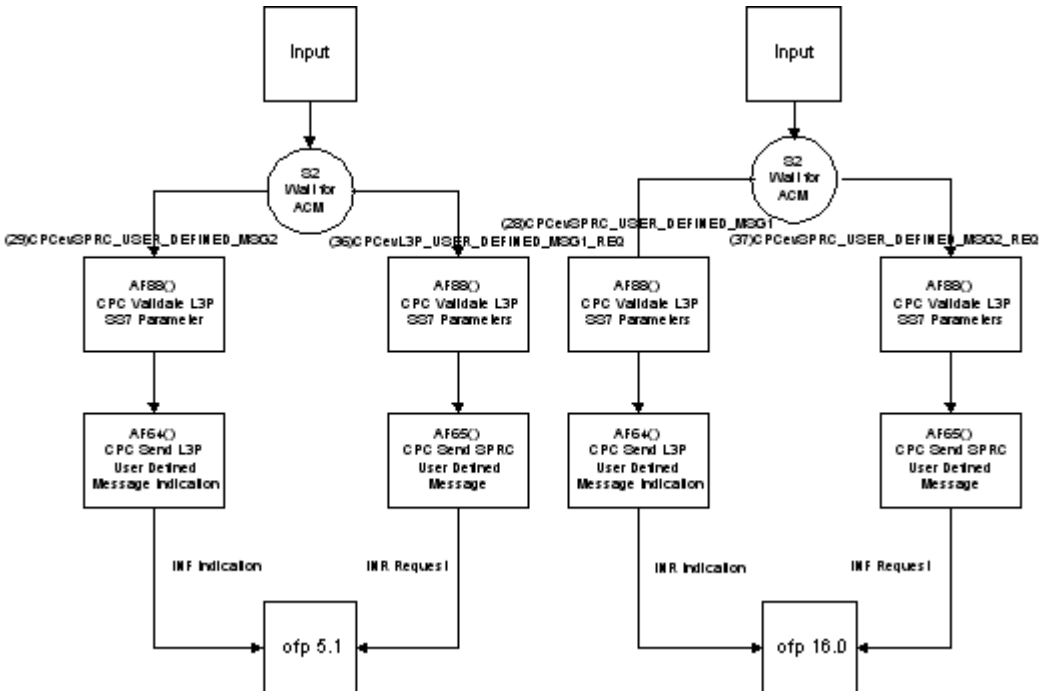


Figure 4-6 L3P CIC User-defined Message Customization



ISUP MTP Pause Logic Options

By default, ISUP processes MTP Pause Indications immediately. ISUP forces all CICs for the paused destination (DPC) out-of-service. Any active calls are listed as the CICs are brought out-of-service. The CICs remain out-of-service until the DPC receives an MTP Resume Indication.

The host can configure the ISUP to delay the processing of the MTP Pause Indication. The host can enable the MTP Pause Logic by setting PPL Configuration Byte 5 in ISUP SPRC. If this flag is set, an MTP Pause Indication is delayed for a period of time determined by an ISUP SPRC PPL Timer. The actual time before a Pause is processed is 10 times the timer value (so setting the timer for 5 seconds causes the Pause expiration in 50 seconds).

During the delay period, all outgoing messages for the DPC are queued in ISUP. If an MTP Resume Indication is received before the timer expiration, the queued messages are sent and the Pause is discarded. If the Pause Timer expires, the queued messages are discarded and the pause is processed as previously described.

Request For Service Message Format

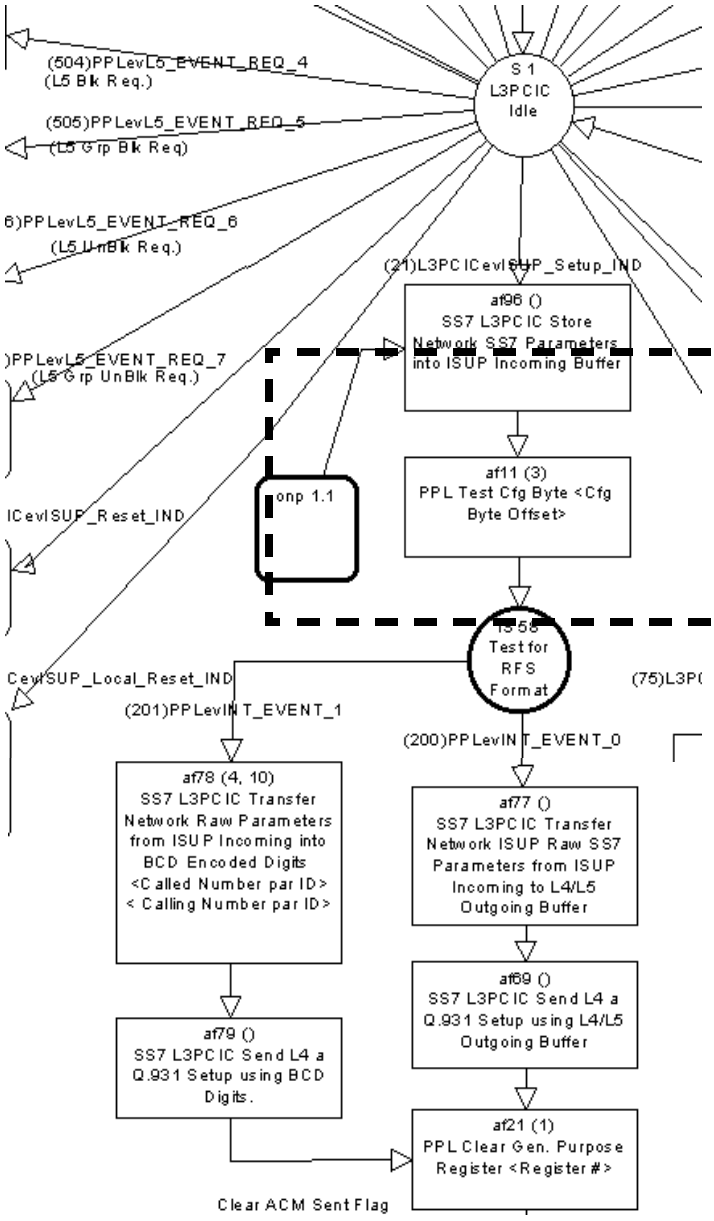
The Called and Calling Party number can be passed to the host as BCD encoded digits, allowing an application to use a generic Feature Group D call model. As shown in *Figure 4-7, Request For Service with BCD*

Encoded Digits (4-104), AF 11 in the L3P CIC state machine tests for the value of L3P CIC Configuration Byte 8 to determine the *Request For Service* message format.

To enable the sending of BCD Encoded digits, the value in Byte 8 must be changed to “BCD Encoded Digits” (0x01) using the *PPL Configure* message. This invokes AF 78, which transfers address data to the ISUP Outgoing buffer as BCD encoded digits.

BYTE	Description	Value
8	Request For Service Format	0 = Raw ISUP (default) 1 = BCD digits

Figure 4-7 Request For Service with BCD Encoded Digits



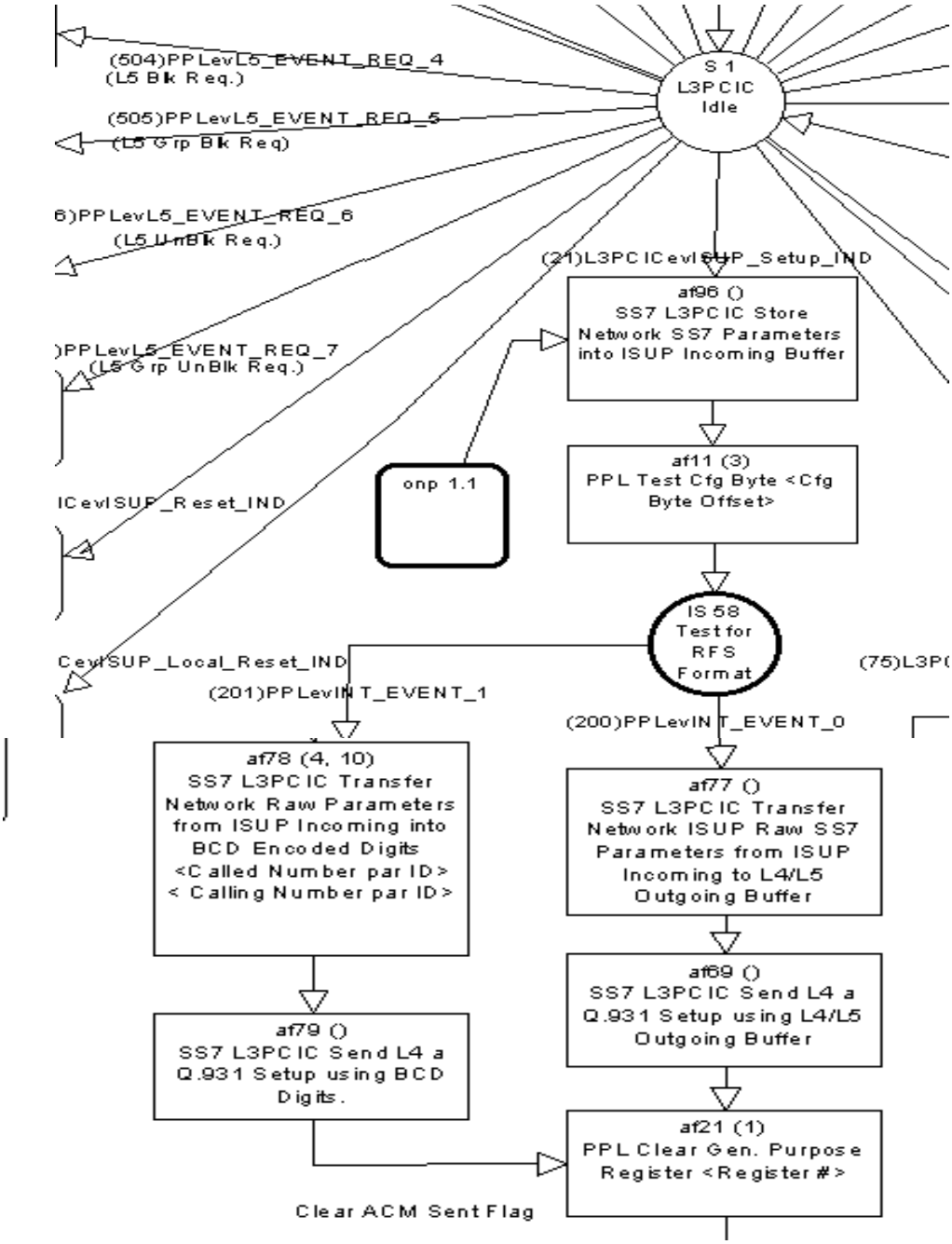
SS7 Parameter Presentation

For incoming calls, the default is to pass all parameters to the host in an SS7 Data ICB message, as shown in *Figure 4-8, Default Parameter Handling (4-105)*. The host can configure the CSP to pass specific parameters, or no parameters, by modifying the L3P CIC state machine using the PPL Tool.

1. Upon reception of an ISUP Setup Indication, AF 96 stores the SS7 parameters into the ISUP Incoming Buffer.
2. AF 11 tests the L3P CIC Configuration Bytes to determine the Request For Service format (Raw ISUP or BCD Encoded Digits).

3. As the default format is Raw ISUP (0), AF 77 is invoked, which transfers all network ISUP parameters into the L4/L5 Outgoing buffer. (To change the *Request For Service* format to BCD-encoded digits, See *Request For Service Message Format (4-102)*.
4. AF 69 sends a Q.931 Setup to L4, including all parameters in the buffer.

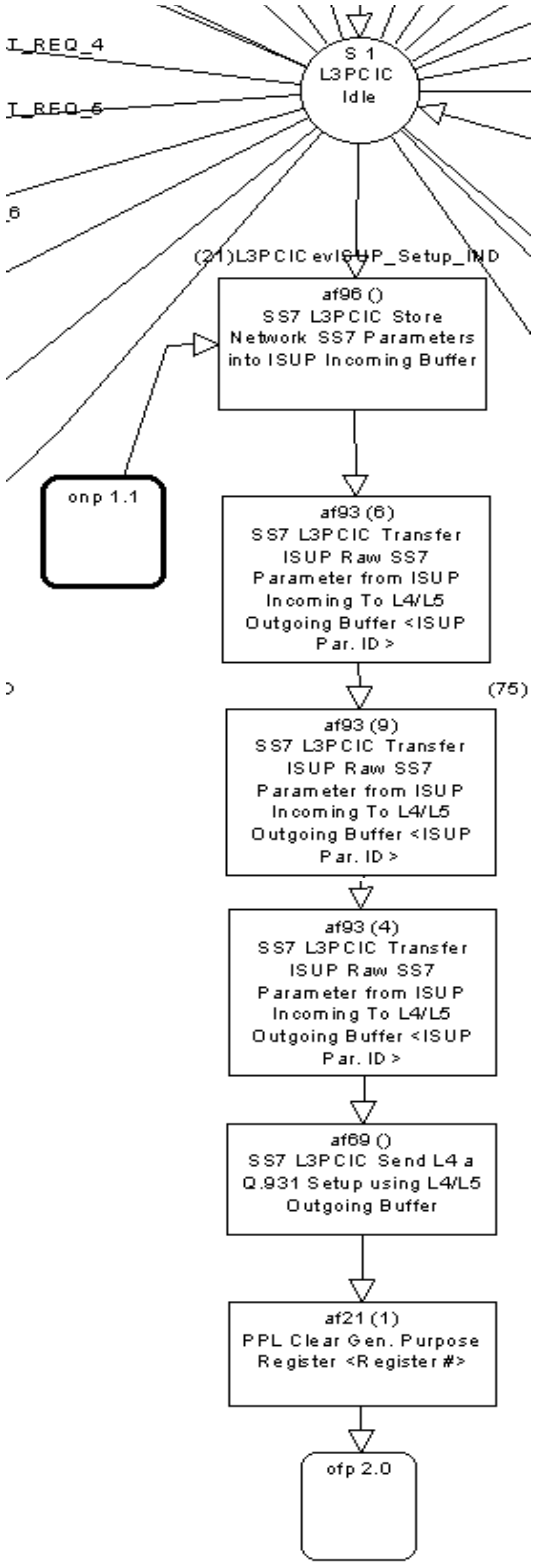
Figure 4-8 Default Parameter Handling



Specified Parameters

The host can configure L3P CIC to send specific parameters using AF 93, which transfers the parameter data in the incoming buffer to the L4/L5 Outgoing Buffer. Argument 1 indicates the Parameter ID. The function must be invoked for each parameter to be sent.

Figure 4-9 Request for Service With Data - Specified SS7 Parameters



No Parameters

For the host to receive a *Request For Service With Data* with no SS7 parameters, L3P CIC should be modified as shown in *Figure 4-9, Request for Service With Data - Specified SS7 Parameters (4-107)*. Upon reception of the ISUP Setup Indication, AF 69 sends a Q.931 Setup to L4. By removing AF 96, no parameters are stored or sent. Because no digits are to be sent, the test for the *Request For Service* format is removed.

Modifying the Format of an ISUP Message

Overview

You modify the format of an ISUP message using the *SS7 ISUP Message Format Configure* message. ISUP messages can be modified by removing or adding optional parameters that are supported in the applicable ANSI or ITU specification. You must include all of the mandatory parameters that are included in the format of an ISUP message.

To find the format of an ISUP message supported by Dialogic, use the *SS7 ISUP Message Query* message.

Finding Parameter IDs

Depending on the telecommunications standard being used, you find the IDs for optional parameters listed in the specifications issued by ITU or ANSI.

Example

The example script file below shows the *SS7 ISUP Message Format Configure* message being used to change the format of the RELEASE message. The *Release* message format includes 13 optional parameters:

- User-to-User Information
- Automatic Congestion Level
- Parameter Capability Information
- Access Delivery Information
- Redirection Number
- Remote Operations
- Network Specific Facilities
- Redirection Information
- Access Transport
- Signaling Point Code
- User-to-User Indicators
- Display Information
- Redirection Number Restriction

Example Configuration Script File

This script file, sent by the host, configures the *Release* message format to include only four optional parameters. This example uses the optional parameter IDs in the ITU-T Q.763 specification.

```
00 20 00 6B 00 00 FF 00 01 08 01 00 04 0C 01 00 01 12 02
   00 01 04 03 01 00 27 01 01 0C 03 0A 20 01 81
```

Example Message Format Table with Values

The table below interprets the information provided in the script file shown earlier in this section. The *Release* message has only one mandatory fixed parameter: Message ID. Because the ISUP Message Index and ISUP Message ID fields indicate this parameter ID, the parameter ID does not need to be repeated in the MFP 1: ID field.

Note:
Maximum Length of 0x00 means infinity.

Byte	Field Description	Value
0	Frame	0xFE
1	Length, MSB	0x00
2	Length, LSB	0x2C
3	Message Type, MSB	0x00
4	Message Type, LSB	0x6B
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0xFF
8	Address Method	0x00
9	Number of Address Elements	0x01
10	Address Type	0x08 (SS7 Stack)
11	Data Length	0x01
12	Data[0] SS7 Stack ID	0x00
13	ISUP Message Index	0x04
14	ISUP Message ID	0x0C
15	ISUP Message Priority	0x01
16	Number of Mandatory Fixed Parameters (MFP)	0x00
17	Number of Mandatory Variable Parameters (MVP)	0x01
18	MVP 1: ID	0x12 (Cause Indicators)
19	MVP 1: Minimum Length	0x02
20	MVP 1: Maximum Length	0x00
21	Optional Parameters Allowed	0x01 (Yes)
22	Number of Optional Parameters	0x04
23	OP 1: ID	0x03 (Access Transport)
24	OP 1: Minimum Length	0x01
25	OP 1: Maximum Length	0x00
26	OP 2: ID	0x27 (Auto. Congestion Level)
27	OP 2: Minimum Length	0x01
28	OP 2: Maximum Length	0x01
29	OP 3: ID	0x0C (Redirection Number)
30	OP 3: Minimum Length	0x03
31	OP 3: Maximum Length	0x0A
32	OP 4: ID	0x20 (User to User Information)

Byte	Field Description	Value
33	OP 4: Minimum Length	0x01
34	OP 4: Maximum Length	0x81
35	Checksum	0x2A

Important ISUP PPL Information

Overview

This section highlights some of the more important ISUP PPL components information related to SS7 customization. Complete Configuration Byte values, and PPL timers and events are documented in the *SS7 PPL Information* of this guide.

Configuration Bytes

This section identifies important PPL Configuration Bytes for each component, and their default values.

ITU L3P CIC

The L3P CIC PPL Configuration Bytes configure the following:

- Bytes 1-3 - CGB, CGU, GRS

  Point to the byte locations of the Circuit Group Block (CGB), Circuit Group Unblock (CGU), and Circuit Group Reset (GRS) parameters, which are used when generating group messages. Other functions in the system access these parameters. You can move the location of the parameter values within the L3P CIC Configuration Bytes; however, you must update Bytes 1-3 to point to their location.

- Bytes 4-7 - SS7 Parameter Information For Outgoing Calls

  Required when the host sends BCD-encoded digits.

- Byte 8 - *Request For Service With Data* message format

  For incoming calls. Address information can be passed as SS7 parameter data or as BCD-encoded digits.

- Byte 9

  Determines if an incoming SS7 call requires a response with the ACM/ANM or CON messages.

- Byte 10 - Host Out-of-service and In-service Operation Flag

  This flag determines if a reset or block/unblock sequence is used for host initiated out-of-service and in-service transitions.

- Bytes 15-145 - Values For SS7 Parameters In Outgoing Messages

  These values are used unless the host overrides them in an SS7 Parameters ICB. Values are included for the following messages:

Message	Value
ACM	Address Complete Message
ANM	Answer Message
BLO	Blocking
CGB	Circuit Group Blocking
CGU	Circuit Group Unblocking
CON	Connect Message
CPG	Call Progress
GRS	Circuit Group Reset
IAM	Initial Address Message
REL	Release
UBL	Unblocking

ITU ISUP CPC

The ISUP CPC Configuration Bytes contain values for the Dual-seizure Control Flag and CPC-initiated REL parameters.

- Byte 1 - Dual-seizure Control Flag

  Upon detection of a dual-seizure condition (collision between incoming and outgoing IAMs); this flag determines which side is dropped.

- Bytes 10–16, 25–31, and 40–46 - parameter values for CPC-initiated Releases, depending on the failure reason

  Under certain circumstances, ISUP CPC must initiate a release to the network. For example, expiration of the T8 timer results in a forward REL with a Cause Value of “Temporary Failure” ITU ISUP SPRC.

ITU ISUP SPRC

- Byte 1 - Range & Status Parameter ID

  This ID is required for logic in ISUP, which identifies group messages and routes them to the correct group state machine based on the value of the range field.

- Byte 2 - Byte Offset of the Range Field

  The range field of the Range and Status parameter is assumed to be one octet. However, its placement in the Range and Status parameter is not assumed. This Configuration Byte entry

contains the byte offset into the parameter for the range field (this is 0 for ITU). This is done, along with the Range and Status parameter name, in order to allow ISUP to be able to handle ISUP message and parameter variants.

- Byte 3 - ITU Service Indicator

  This field is placed in to every outgoing ISUP message as a field in the MSU Service Information Octet.

- Byte 4 - ITU Subservice Field

  This field is placed into every outgoing ISUP message as a field in the MSU Service Information Octet.

- Byte 5 - MTP Pause Logic Flag

  After an MTP Pause indication is received, the SPRC can either process the pause immediately, or it can delay the pause processing and queue any outgoing messages to MTP. If an MTP Resume indication is received before timer expiration, the queued messages are sent to MTP. If the timer expires, the queued messages are discarded and the pause is processed. (For a description of ISUP MTP Pause Logic, see *ISUP MTP Pause Logic Options (4-102)*

ANSI L3P CIC

- Bytes 1 and 3 - CGB, CGU, and GRS

  Point to the byte locations of the Circuit Group Block (CGB), Circuit Group Unblock (CGU), and Circuit Group Reset (GRS) parameters, which are used when generating group messages. These parameters are accessed by other functions in the system. The location of the parameter values can be moved within the L3P CIC Configuration Bytes, however, Bytes 1 and 3 must be updated to point to their location.

- Bytes 4-7 - SS7 parameter information

  Required for outgoing calls when the host sends BCD-encoded digits in an *Outseize Control* message.

- Byte 8 - *Request For Service With Data* message format

  For incoming calls. Address information can be passed as SS7 parameter data or as BCD-encoded digits.

- Byte 10 - Host Out-of-service And In-service Operation Flag

  This flag determines if a reset or block/unblock sequence is used for host initiated out-of-service and in-service transitions.

- Bytes 15-137 - SS7 Parameter Values For Outgoing Messages

  These values are used unless overridden by the host in an SS7 parameters ICB.

ANSI ISUP CPC

The ISUP CPC Configuration Bytes control the Dual-Seizure Control Flag and CPC-initiated REL parameters.

- Byte 1 - Dual-Seizure Control Flag

  Upon detection of a dual-seizure condition (collision between incoming and outgoing IAMs); this flag determines which side is dropped.

- Bytes 10–16, 25–31, 40–46, and 51–61 - parameter values for CPC-initiated Releases, depending on the failure reason.

  Under certain circumstances, ISUP CPC must initiate a release to the network. For example, expiration of the T8 timer results in a forward REL with a Cause Value of “Temporary Failure.”

ANSI ISUP SPRC

ISUP SPRC PPL Configuration Bytes control configuration flags, messages, and other important values, as follows:

- Byte 1 - Range & Status Parameter ID

  This ID is required for logic in ISUP which identifies group messages and routes them to the correct group state machine based on the value of the range field.

- Byte 2 - Byte Offset of the Range Field.

  The range field of the Range and Status parameter is assumed to be one octet. However, its placement in the Range and Status parameter is not assumed. This Configuration Byte entry contains the byte offset into the parameter for the range field (this is 0 for ANSI). This is done, along with the Range and Status parameter name, in order to allow ISUP to be able to handle ISUP message and parameter variants.

- Byte 3 - ANSI Service Indicator

  This field is placed in to every outgoing ISUP message as a field in the MSU Service Information Octet.

- Byte 4 - ANSI Subservice Field

  This field is placed into every outgoing ISUP message as a field in the MSU Service Information Octet.

- Byte 5 - MTP Pause Logic Flag

  Upon reception of an MTP Pause indication, the SPRC can either process the pause immediately, or it can delay the pause processing and queue any outgoing messages to MTP. If an MTP Resume indication is received before timer expiration, the queued messages are sent to MTP. If the timer expires, the queued messages are discarded*ISUP MTP Pause Logic Options (4-102)***.**

- Bytes 6-26 - SPRC-initiated Confusion

  CFN messages to the network in response to an unrecognized or corrupt message.

- Bytes 30-32 - parameter values for an Unequipped CIC

  UCIC message to the network in response to a network message that pertains to an unknown or out-of-service CIC.

L3P CIC PPL Configuration Bytes

The mapping of the PPL Configuration Bytes for the L3P CIC component allows you to move Configuration Bytes for a message without modifying the PPL protocol.

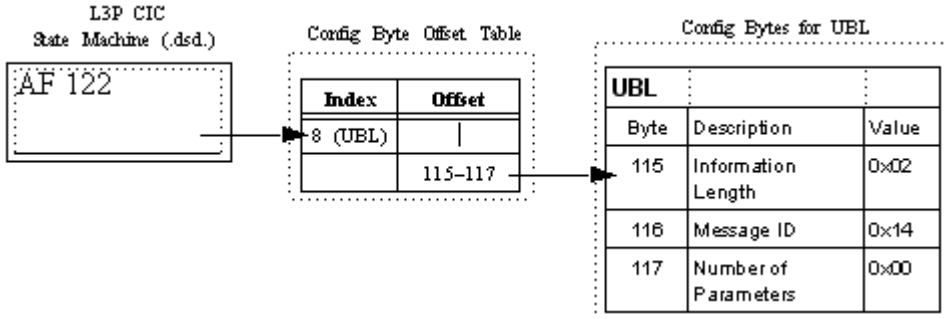
Existing atomic functions point to specific Configuration Bytes for the format information on a specific message. If this information was moved to a new Configuration Byte location, the DSD would have to be updated to reflect the new location of the Configuration Bytes.

There are three blocks of 200 Configuration Bytes each, for a total of 600 Configuration Bytes. Bytes 501–600 are used as the Configuration Byte Offset Table, which contains an index pointing to the location of the Configuration Bytes with the format information for each message.

Atomic Function 122 in the L3P CIC state machine points to the index in the Configuration Byte Offset Table, which indicates the location of the Configuration Bytes for a specific message. If you move the location of the Configuration Bytes for a message, you only need to update the Configuration Byte Offset Table, using the *PPL Configure* message, to reflect the new location.

The next figure shows the usage of the Configuration Byte Offset Table by the L3P CIC state machine to access Configuration Bytes for a specific message. See *SS7 PPL Information* for ITU and ANSI Configuration Byte offset values.

Figure 4-10 Use of Configuration Byte Offset Table



Protocol Timers

Each PPL component has multi-purpose timers, which you can activate at any time. Each component using timers has a table that contains information on specific timers (name, value). When a protocol requires a timer, an atomic function is initiated to activate one of the PPL timers and to point to an index in the components timer table, which contains the value for the required timer. See *SS7 PPL Information* in this manual for default timer values.

Other Customization

The table below lists other common SS7 customization which is implemented through modification of PPL Configuration Bytes with the *PPL Configure* message.

To modify...	**Use...**	
	Component	**Config Bytes**
Host-initiated Out-of-Service Logic Flag	L3P CIC	10
Dual-seizure (Glare) Control Flag	ISUP CPC	1
Default Outgoing SS7 Parameters	L3P CIC	All
Service Indicators	ISUP SPRC	3
Subservice Field	ISUP SPRC	4
Range and Status Parameter ID	ISUP SPRC	1
CFN Parameters- ANSI only	ISUP SPRC	All
ISUP CPC Initiated Release Parameters	ISUP CPC	All
ITU CON/ANM/ACM Control	ITU L3P CIC	9
Transmitted LSSU Status Field	MTP2 TXC	All
Network Indicators	MTP3 HMDT and MTP3 HMRT	All

Overlap Call Digit Collection

Overview

Overlap Call Digit Collection provides the capability for overlap calls to collect digits from multiple messages before forwarding them to the host, a capability normally reserved for En-bloc calls.

This feature supports the collection of digits received in a Subsequent Address Messages (SAM) and the inclusion of those digits in a single Request for Service (RFS) message sent to the host. Using this method, the RFS with Address Data is not sent to the host until the SETUP Initial Address Message (IAM) and all the SAMs have been received. Thus, the host is able to process an overlap call as though it were actually En-bloc.

The existing default method of digit collection, by contrast, sends an RFS with Address Digits and a partial Called Party number, followed by SAMs, which are represented as PPL Event Indications. The host has to wait for the additional digits to be processed before continuing its call setup.

This ITU-T variant provides the following expanded functionality:

- The call setup process is made more efficient.
- The number of API messages that must be exchanged between the switch and the host to set up a call are considerably reduced, which improves flow where there are large volumes of traffic.
- This ISUP enhancement is backward compatible with previous versions of ISUP.

See the call flows on the following pages for descriptions of existing and new digit collection and for inter-digit timer events.

Configuration

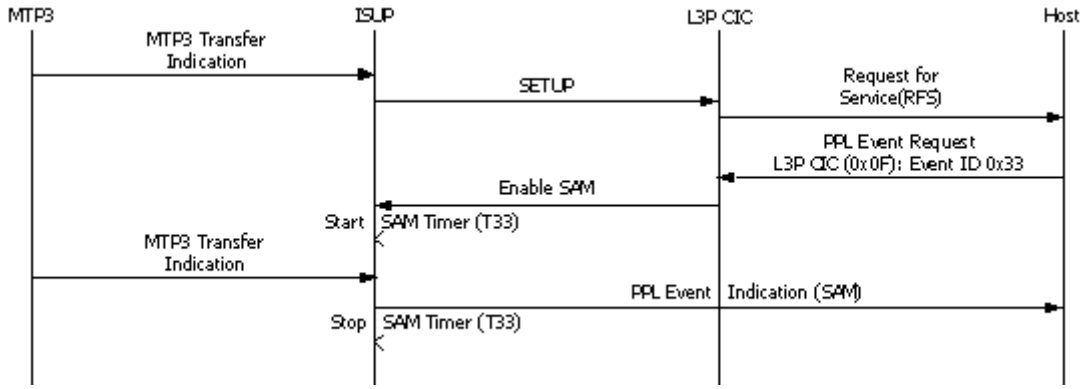
The following configuration applies to the PPL information found in *SS7 PPL Information*:

- The host uses the following two configuration bytes to enable or disable this feature. Both must be enabled:
  - Component 0x0012 (CPC), Config Byte 0xD2
  - Component 0x000F (L3P CIC), Config Byte 0x30
- The host configures the maximum number of digits the CSP must receive before it sends the RFS message to the host using a configuration byte as well:

- Component 0x000F (L3P CIC), Config Byte 0x31 (default is 0x0a)

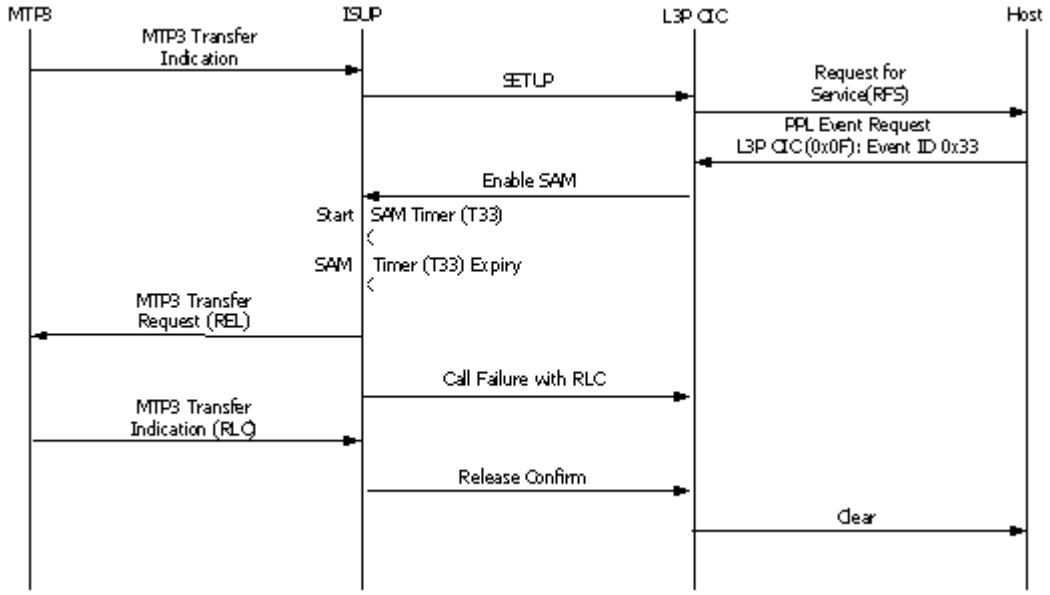
- The host configures the PPL Timer used to determine how long a SAM should be waited for before processing the RFS using the PPL Timer Configuration message
  - Component 0x000F (L3P CIC), Timer ID 0x05

Call Flows Successful Handling of SAM in the Existing Implementation



1. A SETUP request is received with an insufficient number of digits.
2. The host sends a *PPL Request* to L3P CIC to enable the SAM Timer (T33) and wait for SAM.
3. When SAM is received, the SAM Timer (T33) is stopped and a SAM *PPL Event Indication* is sent to the host.

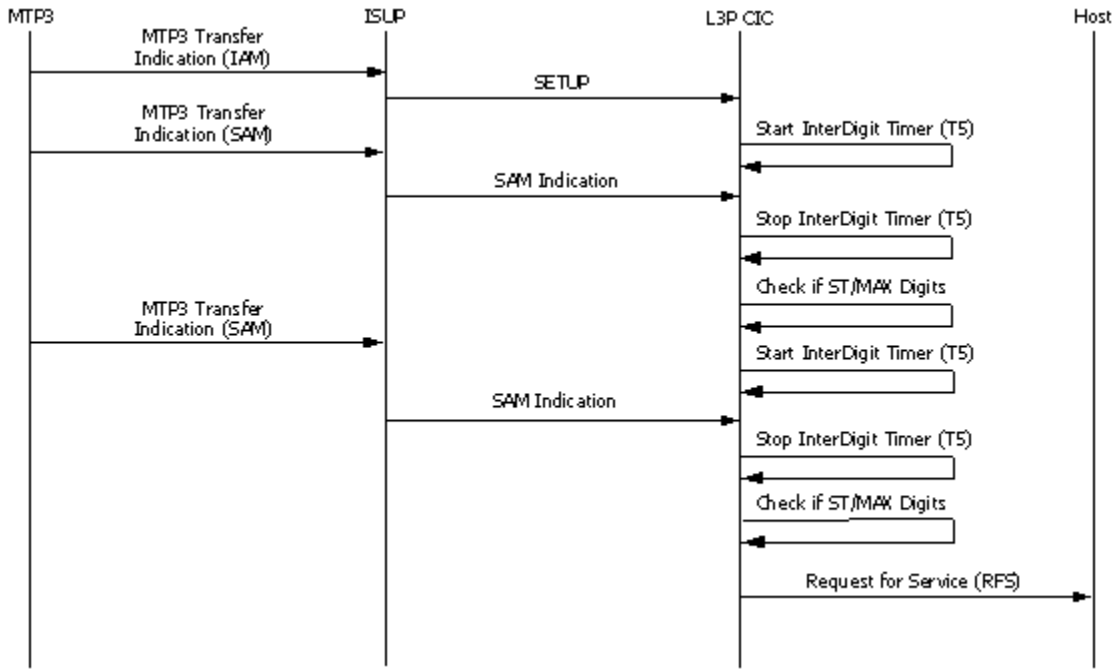
Expiry of SAM Timer in the Existing Implementation



1. A SETUP request is received with an insufficient number of digits.
2. The host sends a *PPL Request* to L3P CIC to enable the SAM Timer (T33) and wait for SAM.
3. If SAM is not received and the SAM Timer (T33) expires, a Call Failure *PPL Event Indication* is sent to L3P CIC, which then sends a Clear message to the host.

Successful Handling of SAM in the New Implementation

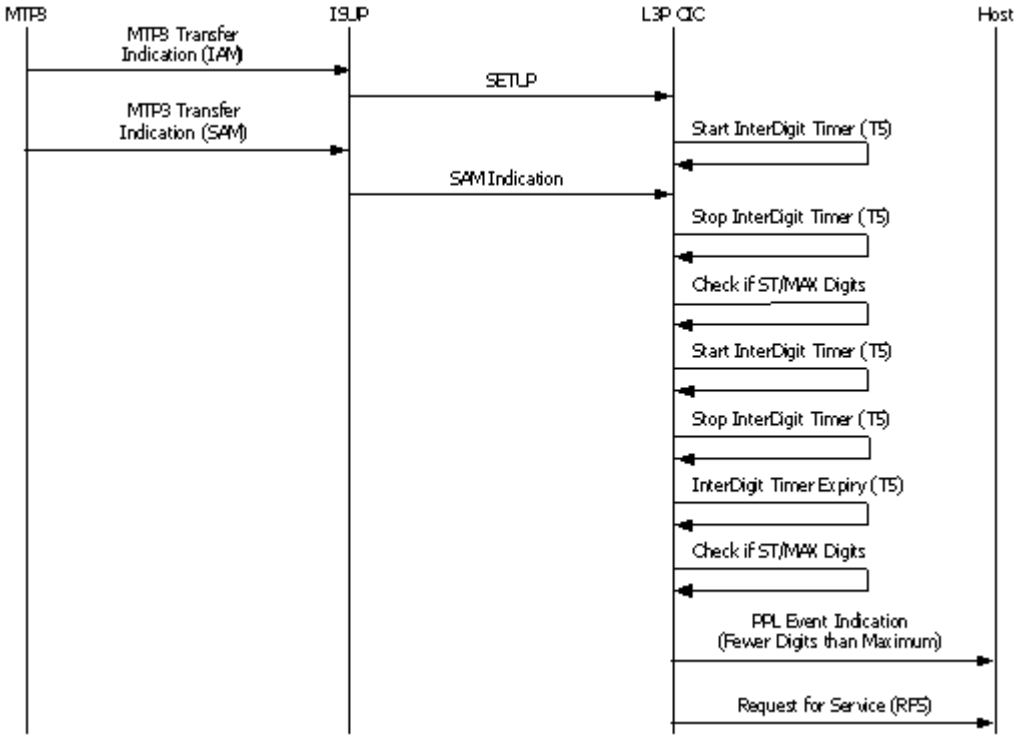
This call flow shows how the collection of digits is handled in SAM after the new functionality has been added.



1. A SETUP request is received with an insufficient number of digits for successful call translation.
2. The InterDigit Timer (T5) is started to wait for a SAM message.
3. If a SAM message is received before the expiry of the InterDigit Timer (T5), the timer is stopped. The digits collected are checked for the presence of the “End of dialing/digits” signal [i.e., the Signal Terminator (ST) default, set to 0x0F] which indicates that no more digits need to be received.
4. If the ST signal is not present in the digits of the SAM message received, or if the maximum number of digits is not received, the InterDigit Timer (T5) is started again.
5. **Only** on receipt of a SAM message containing the ST signal, an RFS containing the complete set of digits is sent to the host.

Important! While the SAM digit collection is occurring, if the host attempts to seize the circuit to make an outgoing call, the message will be NACKed.

Expiry of SAM Timer in the New Implementation



1. A SETUP request is received with an insufficient number of digits for successful call translation.
2. The InterDigit Timer (T5) is started to wait for a SAM message.
3. If a SAM message is received before the expiry of the InterDigit Timer (T5), the timer is stopped. The digits collected are checked for the presence of the ST signal, which indicates that no more digits need to be received.
4. If the ST signal is not present in the digits of the SAM message received, the InterDigit Timer (T5) is started again.
5. But if repeated SAMs do not contain the ST signal, or the number of digits received does not equal the maximum number of digits, the InterDigit Timer (T5) expires and an RFS is sent to the host containing the number of digits received in the IAM and SAM messages.

Automatic Congestion Control

Overview

Automatic Congestion Control (ACC) allows the CSP to handle the following:

- local overloads
- remote overloads

Local overload is the condition where the local SS7 stack on the CSP is overloaded. All incoming calls from the network and all Outseize Control requests from Layer 4 are throttled (rejected) and the CSP comes back to a load level to process steady call traffic.

Remote overload is the condition where a remote node is overloaded or there is a network congestion to the remote node. All Outseize Control requests from Layer 4 get throttled which brings that node to a lower load level.

In either case, for Outseize Control requests, Layer 4 receives an Access Denied Reason of 0x51 when the SS7 is overloaded locally or there is a remote overload at the destination of the call. Layer 4 releases the incoming side with the cause code of Congestion with ACL2.

Configuration

PPL Components

You use the following PPL Components to implement this feature:

- ACC 0xA8 - Configuration Bytes and PPL Timers
- SPRC 0x13 - Configuration Bytes and PPL Timers

In most cases, the parameter's default value is appropriate but depending on the network, you might have to change some parameters.

Refer to *ISUP SPRC (0x0013) (7-61)* for configuration information.

Acceptance Rate and Automatic Congestion Level Values

The Automatic Congestion Level (ACL) values define the two overload levels that you can assign to a destination. For example you would assign a destination to have ACL Level 1 or ACL Level 2 overload.

The Acceptance Rate defines how many calls per second each ACL can handle before the calls are throttled.

You include the Acceptance Rate TLV (0x4E24) in the *PPL Event Request* (0x0044) API message to change the Acceptance Rate for each destination for either ACL levels. The Acceptance Rate is the number of call per second associated with the ACL.

Refer to the *TLV Chapter* in the *API Reference*.

Alarms

Refer to the *Alarm* (0x00B9) API message for information on four general alarms associated with this feature:

- SS7 Signaling Network Congestion (0x0E)
- ISDN/SS7 Signaling Stack Congestion (0x16)
- Signaling Destination Congestion (0x3A)
- Signaling Link Congestion (0x3B)

State Syncing on PPL Event Request for ANM

Terminology

The following terminology appears in this section.

- L3P CIC - Layer 3 Protocol Circuit Identification Code
- L3P BT IUP - Layer 3 Protocol British Telecom Interconnet User Part
- L4CH - Layer 4 Channel

Feature Overview

Prior to this feature, when the PPL Event Request for ANM is sent either to L3P CIC component of ANSI/ITU or to L3P BT IUP component the L4CH SM is not in the answered state. To put the L4CH SM in answered state, you needed to use a custom PPL.

This feature puts the L4CH SM in an answered state when the PPL Event Request for ANM is sent to either of the following:

- L3P CIC component of ANSI or ITU
- L3P BT IUP component for BT IUP

Enabling this Feature

This feature is disabled by default. Follow the steps below to enable it.

ANSI and ITU Protocols

In the L3P CIC component, set configuration byte 0xA7 to 0x01.

Use the *PPL Configure* (0x00D7) message to enable this feature as follows:

```
00 12 00 d7 00 00 FF 00 01 0d 03 00 01 01 00 0f 01 01 a7
   01
```

BT IUP Protocol

In the L3P BT IUP component, set configuration byte 0x3F to 0x01.

Use the *PPL Configure* (0x00D7) message to enable this feature as follows:

```
00 12 00 d7 00 00 FF 00 01 0d 03 00 01 01 00 11 01 01 3f
   01
```

Querying

Use the *PPL Query* message (0x00DE) to query the PPL configuration bytes above.

ANSI / ITU

```
00 0f 00 DE 00 00 ff 00 01 0d 03 00 01 01 00 0f 01
```

BT IUP

00 0f 00 DE 00 00 ff 00 01 0d 03 00 01 01 00 11 01

PPL Information Refer to components 0x000F and 0x0011 in *SS7 PPL Information (7-1)*.

5 TUP, BT IUP & SSUTR2

Purpose

This chapter provides information about the Telephone User Part (TUP), which is a module within the SS7 stack that defines the telephone signaling necessary for the processing of international and national calls. Like ISUP, the term “user” in the context of TUP refers to any functional entity that uses the transport capability provided by SS7’s Message Transfer Part (MTP). TUP is flexible enough to handle most required national and international parameters. The Dialogic TUP interface includes the ability for the host to manage the TUP Virtual CIC call control and circuit management. The Dialogic TUP includes the SSUTR2 variant which is used to define the signaling interface between a fixed network and a mobile network.

More Information

See the Appendix at the end of this book for tables of TUP-supported messages, Field IDs for supported fields and restrictions on the implementation of TUP messages.

Introduction to TUP Call Control

Overview

The host manages TUP call control and circuit management through the following mechanisms:

- Common Call Control API messages, using Formatted Fields (TUP) ICBs to pass data.
- Channel State

  DS0 Status Change messages notify the host of service state changes and channel purges. The *Service State Configure* message allows the host to bring channels in and out-of-service.
- The *PPL Event Indication* and *PPL Event Request* messages allow the host to send and receive messages to and from PPL components. See *SS7 PPL Information* for events supported per component.

The CSP sends *PPL Event Indication* messages to the host in response to various SS7-related call processing events, such as reception of an ACM, ANN, ANU, ANC in the case of TUP, and ACF (ACT/ACS/ ACP/ALT/ALS/ALP) and RIU in the case of TUP SSUTR2.

,Data is included in the SS7 TUP Formatted Fields ICB. If you do not require this information, Acknowledge (ACK) the message and then ignore it.

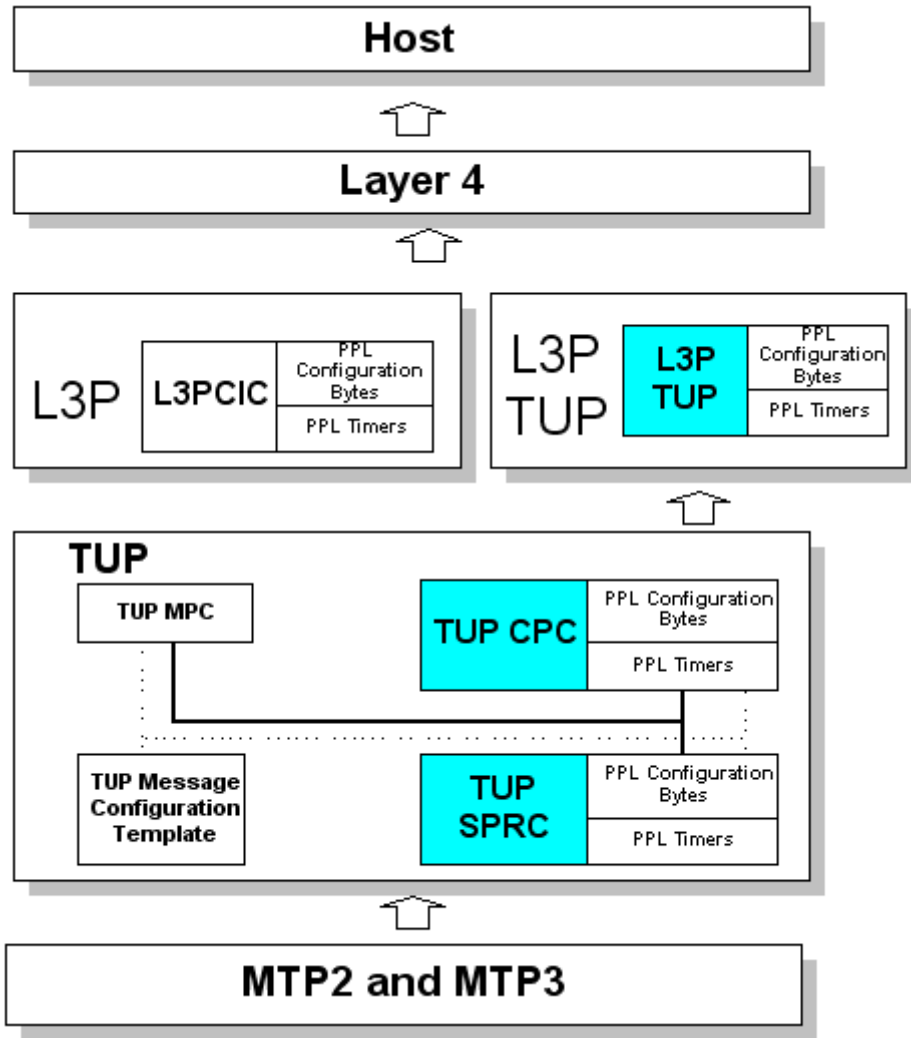
The host can generate SS7-related call processing or management events using the *PPL Event Request* message.

You can modify the default call model to have the Called and Calling Party Numbers passed to the host as BCD encoded digits. *SCCP/TCAP (6-1)* for more information.

TUP Incoming Call Setup

Overview This section describes TUP incoming call setup.

Diagram The diagram illustrates the default interaction between the host and the SS7 PPL components during an incoming SS7 call using TUP. Shading identifies the software components involved in the call.



Call Sequence The default incoming call sequence is described in the table below. To customize the presentation of an incoming call, See *SCCP/TCAP (6-1)*

Stage	Description
1	Raw TUP messages are received by MTP and passed to TUP.

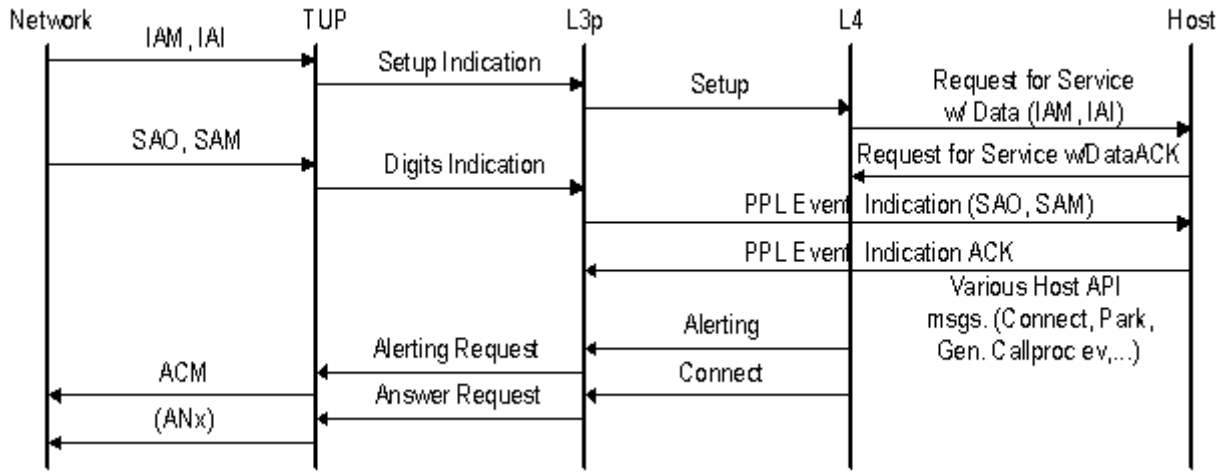
Stage	Description
2	TUP SPRC is responsible for routing the data to the appropriate component. Assuming the data is for a valid CIC, it is routed to TUP CPC for processing. If the data is maintenance related, it is routed to MPC.
3	Assuming no error conditions occur, TUP consults the TUP Message Configuration Template to identify the message and translate the raw data into the format of an SS7 TUP Formatted Fields ICB to pass to L3P TUP. To modify the TUP message formats, the host issues the TUP Message Configuration message. The host can configure which parameters are included and also customize messages for TUP variants.
4	L3P TUP passes generic setup indication messages to Layer 4, which drive Layer 4 into an inseized state and generate a Request For Service With Data message to include an SS7 TUP Formatted Fields ICB with all received parameters to the host. You can modify this format to specify the parameters passed or to have the Called and Calling Party passed to the host as BCD encoded digits.

TUP Incoming Call Flows

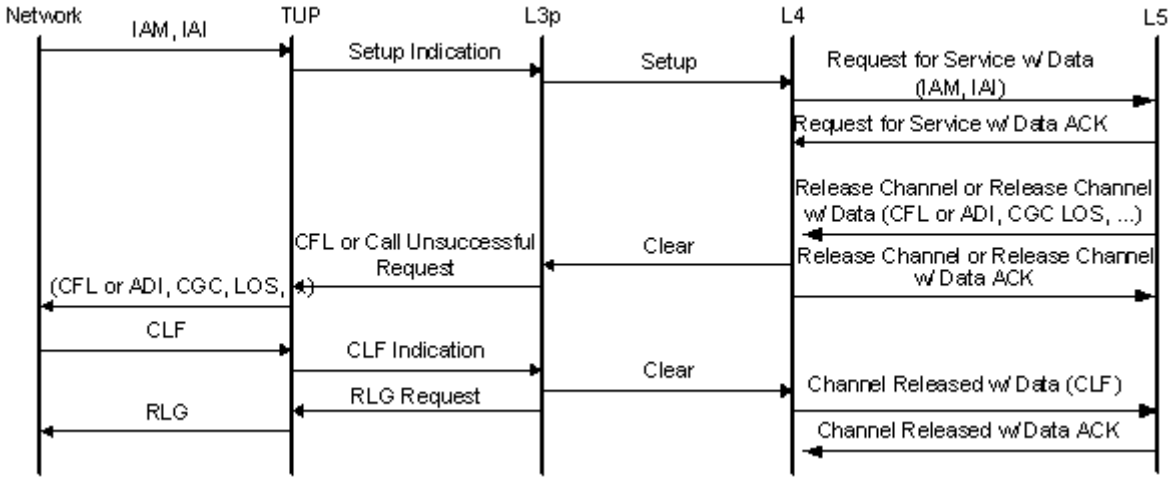
This section includes call flows for incoming call handling and release. Parenthesis in Layer 5 messages indicate data for the TUP message of the specified type.

ANx = ANC, ANN, or ANU

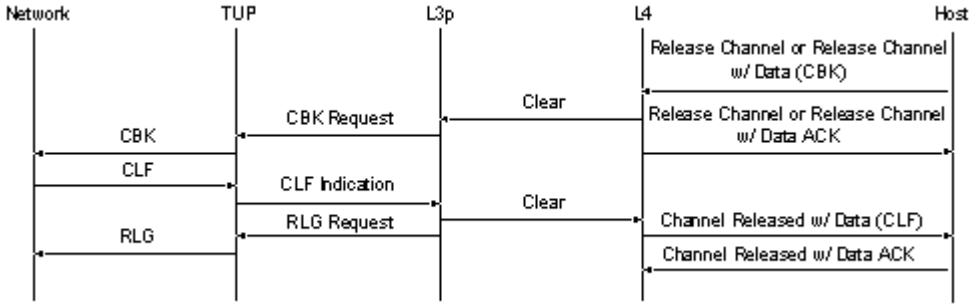
Incoming Call (Not Controlling Exchange) - Basic Call Setup



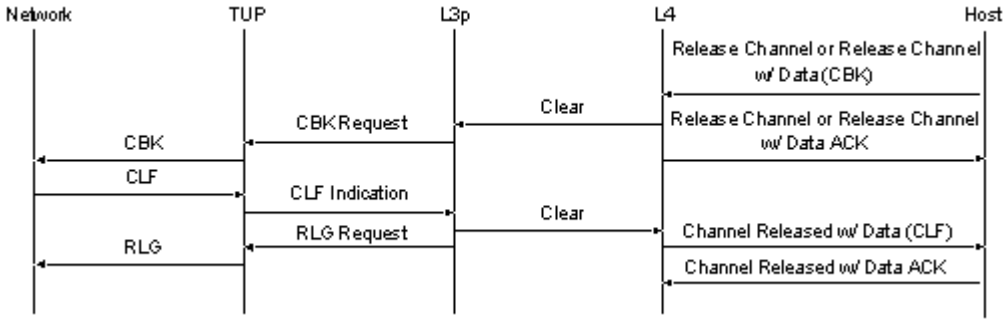
Incoming Call (Not Controlling Exchange) - Call Failure



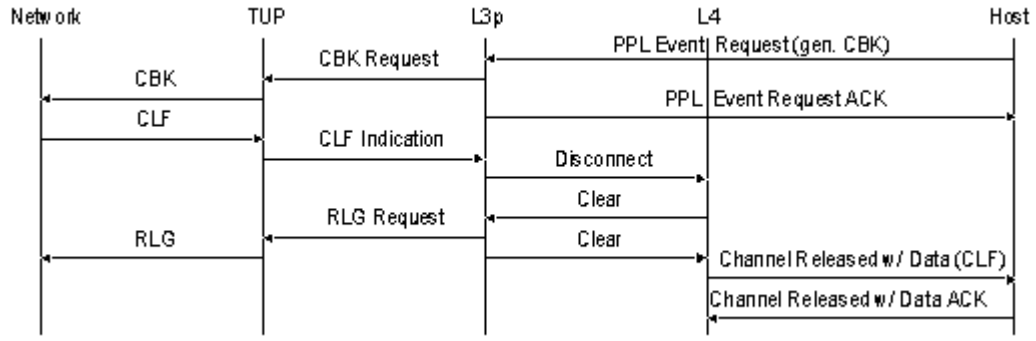
Incoming Call (Not Controlling Exchange) - Remote Release



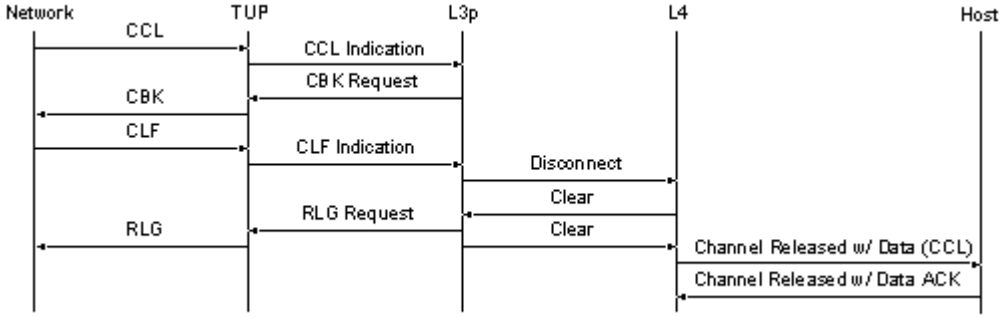
Incoming Call (Not Controlling Exchange) - Local Call Release



Incoming Call (Not Controlling Exchange) - Local Call Clear-back without Release

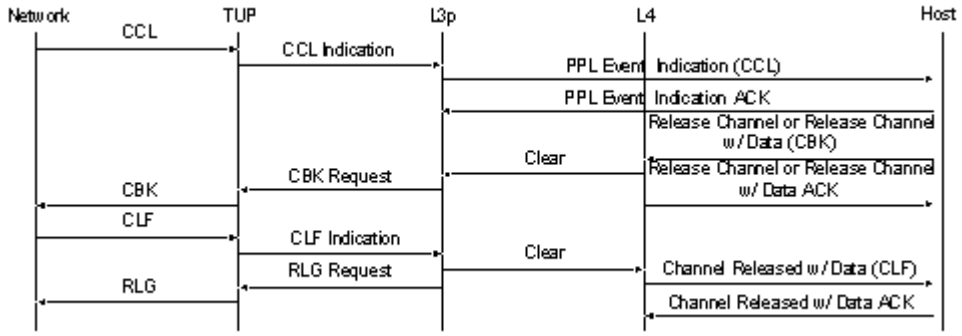


Incoming Call (Controlling Exchange) - Remote Call Calling Party Clear Option 1 (default)



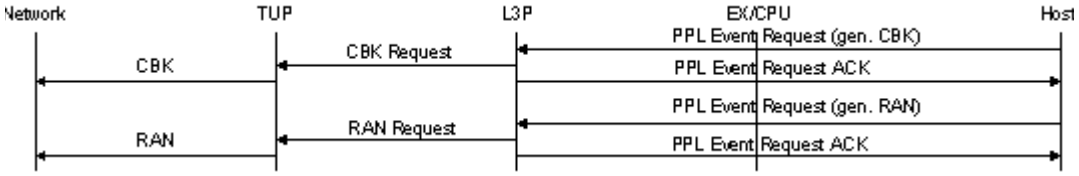
Important! In this call flow, an incoming CCL results in the automatic tear down of the call (generation of CBK).

Incoming Call (Controlling Exchange) - Remote Call Calling Party Clear Option II

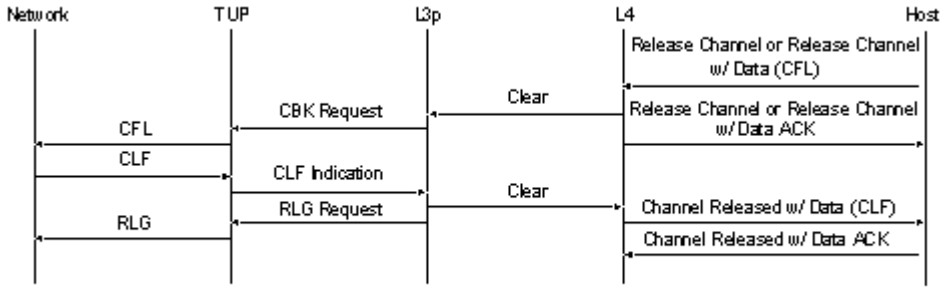


Important! For this call flow, you must enable Config Byte 2 of the L3P TUP component to receive a *PPL Event Indication* (CCL).

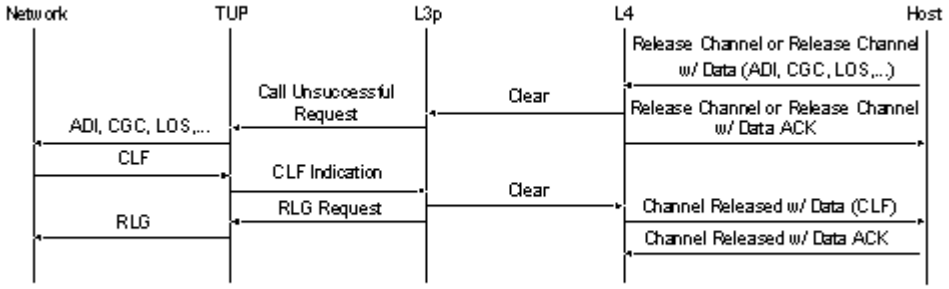
Incoming Call (Not Controlling Exchange) - Local Call Clear-back with Re-answer



Incoming Call (Not Controlling Exchange) - Local Call Failure (Before Answer)



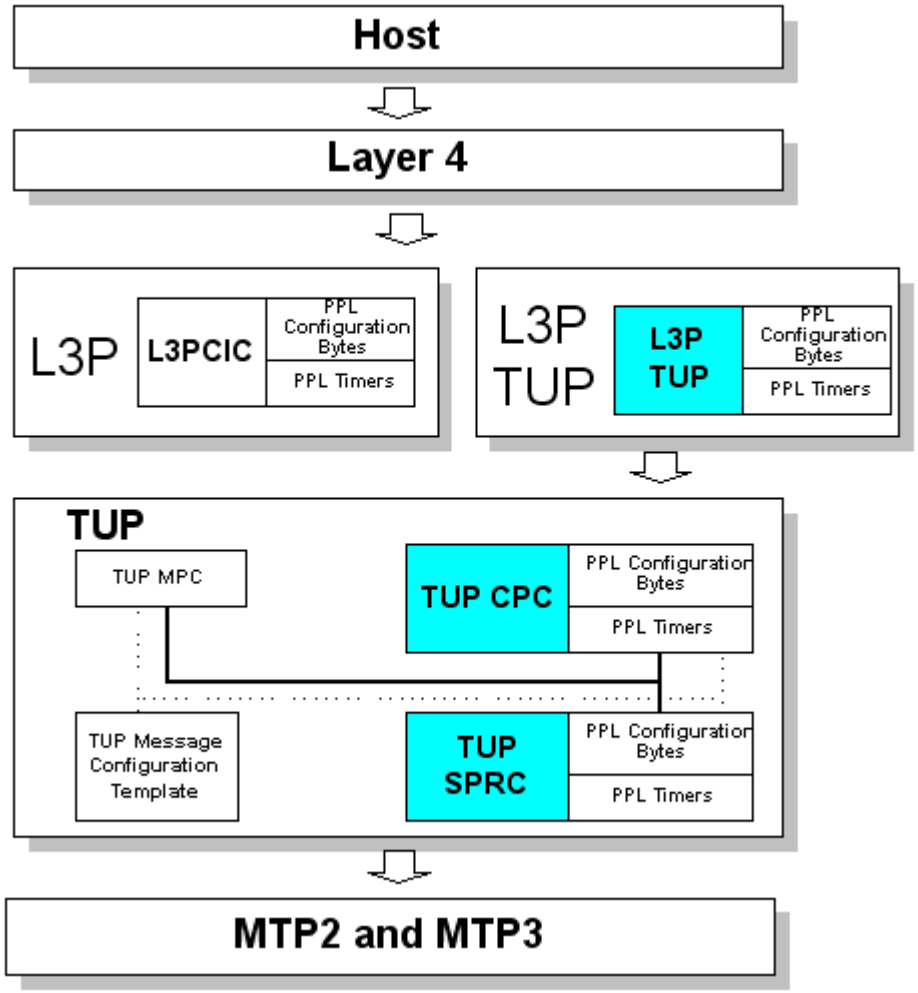
Incoming Call (Not Controlling Exchange) - Local Call Unsuccessful (Before ACM)



TUP Outgoing Call Setup

Overview This section describes TUP outgoing call setup.

Diagram This diagram illustrates the interaction between the host and the SS7 PPL components in the handling of an outgoing SS7 call. Shading identifies the software components involved in the call.



The Default Outgoing Call Sequence The default outgoing call sequence is described in the table below. To customize the presentation of an incoming call, see *SS7 Customization with ISUP Messages (4-95)*.

Stage	Description
1	An outgoing call is initiated by the host with the Outseize Control message. The CSP will accept address data as either BCD-encoded digits or in an SS7 TUP Formatted Fields ICB. The Outseize Control message drives L4 into an outseize state and an outseize request is sent to L3P TUP. NOTE: The Outseize Instruction List Configure message is not supported for SS7. All outseize instructions must be included in the Outseize Control message.
2	L3P TUP translates the L4/L5 event into an TUP event. Any fields sent from L4/L5 are concatenated with pre-stored field data in the PPL Configuration Bytes.
3	The fields and events are sent to the appropriate TUP Component. If field validation is successful, the fields are formatted into a raw SS7 message and sent to TUP SPRC. If validation fails, a PPL Event Indication message of Protocol Violation is sent to the host.
4	TUP SPRC routes the message to MTP for transmission.

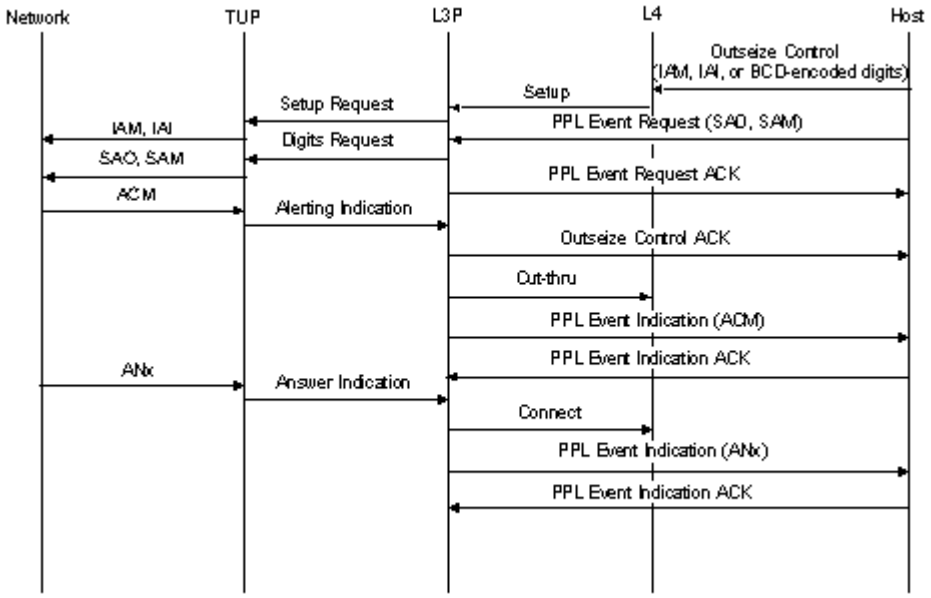
TUP Outgoing Call Flows

This section includes call flows for outgoing call handling and release.

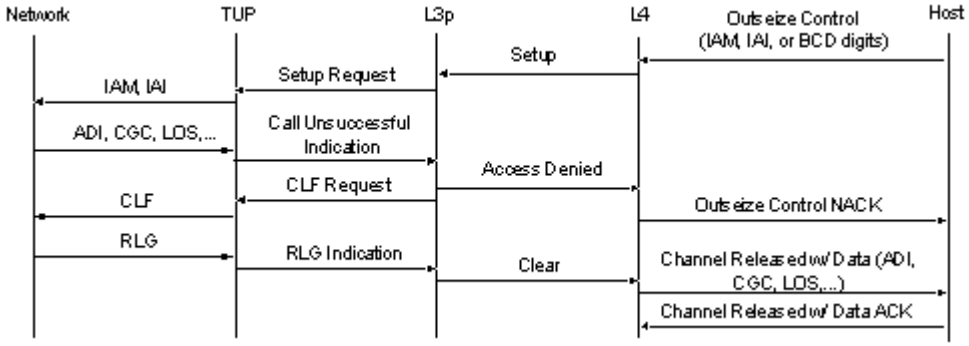
ANx = ANC, ANN, or ANU

The *PPL Event Indication* messages indicated by an asterisk (*) are not enabled by default. To enable indication of ACM, set Config Byte 1 of the L3P TUP component to 0x01. To enable indication of ANC/ANN/ANU set Config Byte 8 to 0x01.

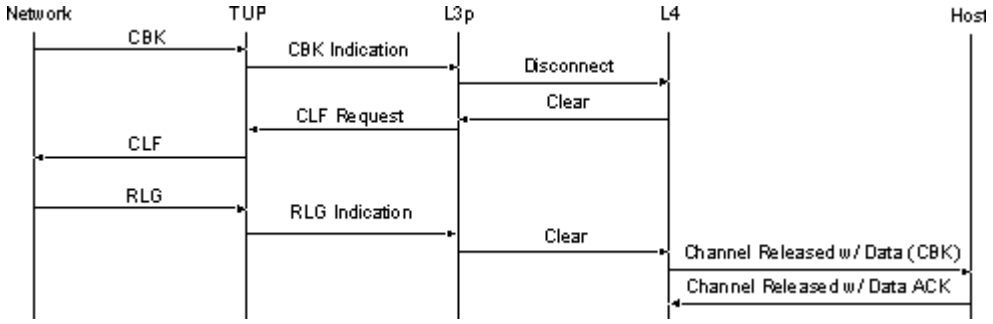
Outgoing Call (Controlling Exchange) - Basic Call Setup



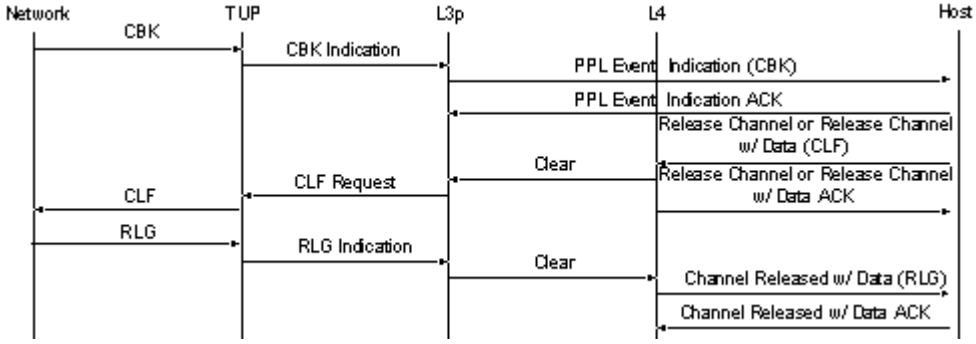
Outgoing Call (Controlling Exchange) - Call Unsuccessful



Outgoing Call (Controlling Exchange) Remote Clear-back Option1 (Default)

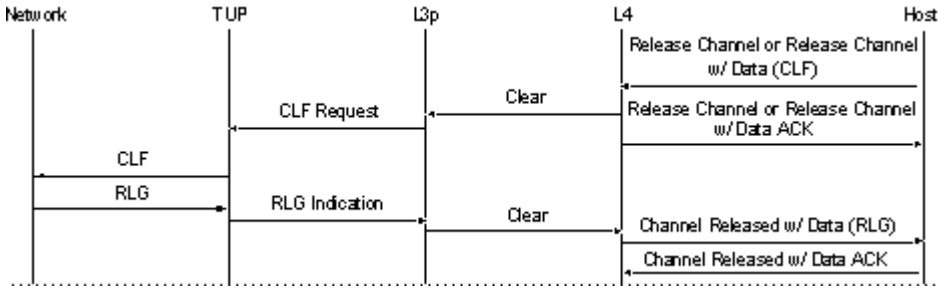


Outgoing Call (Controlling Exchange) Remote Clear-back Option II (Manual Handling)

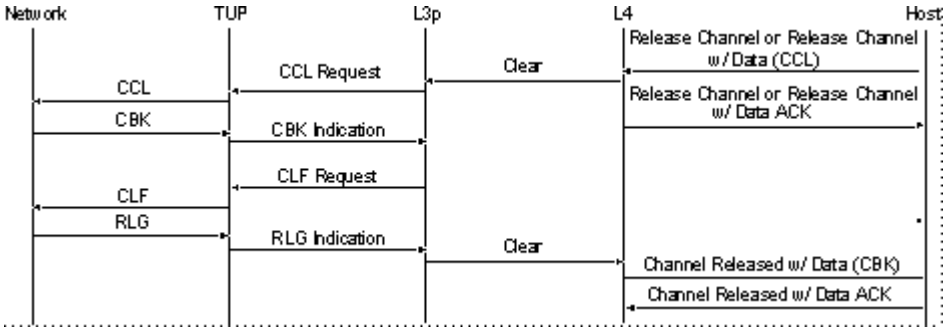


Important! For this call flow, you must enable Config Byte 5 of the L3P TUP component to receive a *PPL Event Indication* (CBK).

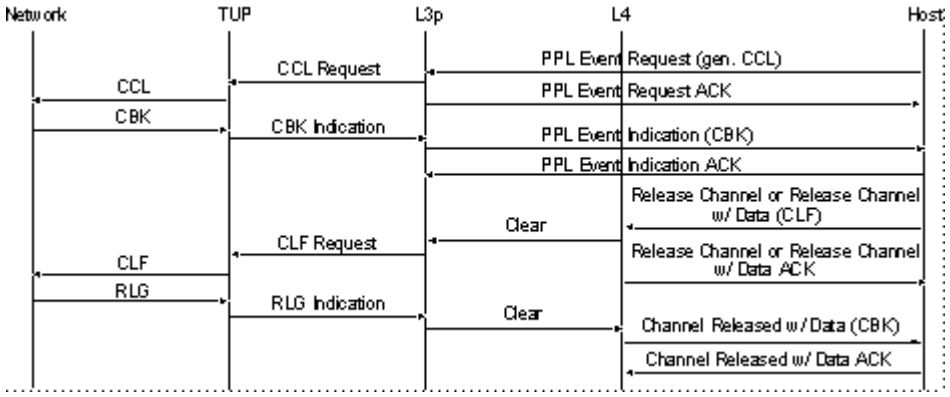
Outgoing Call (Controlling Exchange) - Local Release



Outgoing Call (Not Controlling Exchange) - Local Calling Party Option I (Automatic Handling, Default)



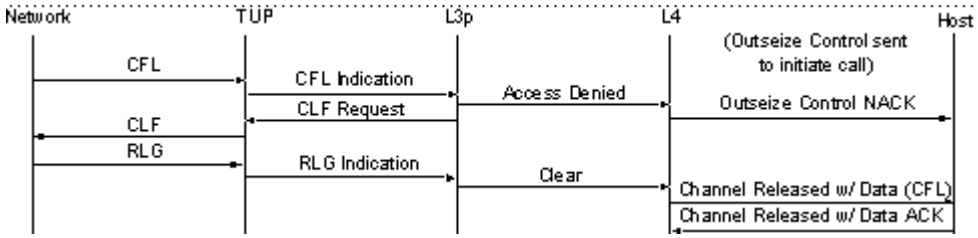
Outgoing Call (Not Controlling Exchange) - Local Calling Party Clear Option II (Manual Handling)



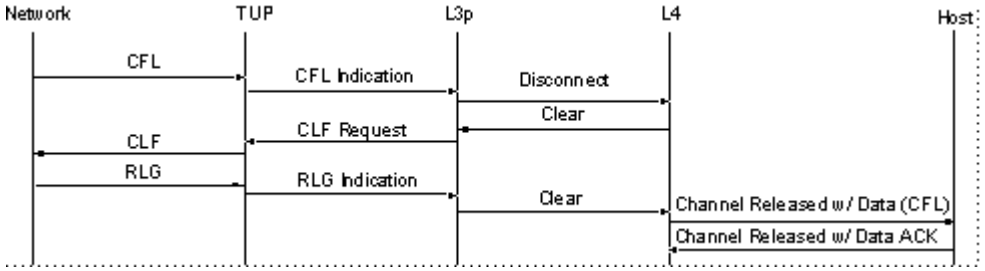
Important! For this optional call flow, you must:

- Enable Config Byte 5 of the L3P TUP component to receive a *PPL Event Indication* (CBK).
- Send a *PPL Event Request* (CCL) message.

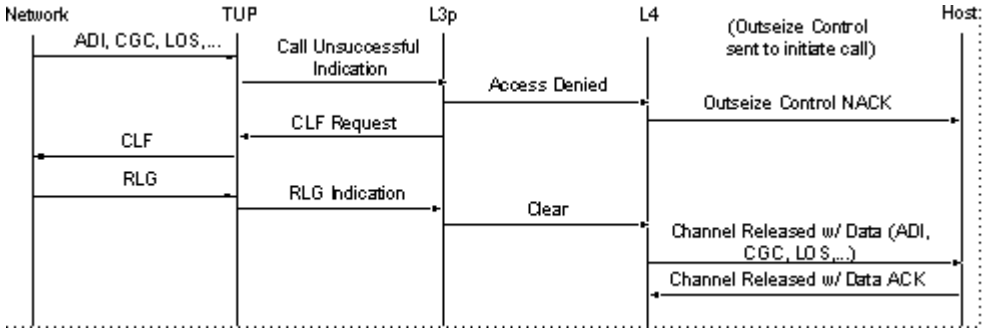
Outgoing Call (Controlling Exchange) - Remote Call Failure before ACM



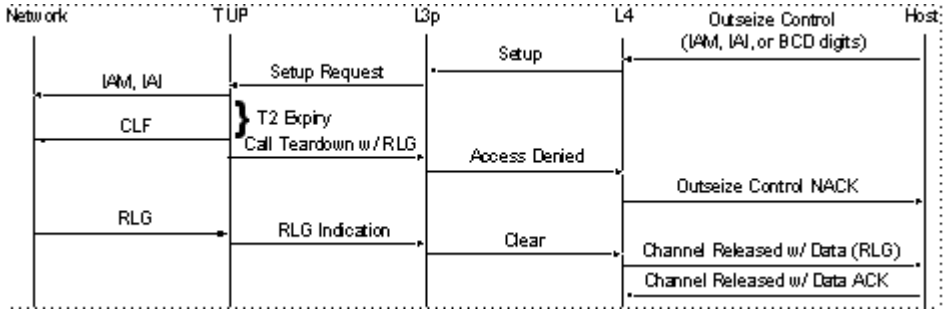
Outgoing call (Controlling Exchange) - Remote Call Failure after ACM



Outgoing Call (Controlling Exchange) - Remote Call Unsuccessful



Outgoing Call (Controlling Exchange) - T2 (Wait for ACM) Expiration



TUP Virtual CIC

Overview

The TUP Virtual CIC is a protocol configuration that enables application systems to use and manage the functions of China SS7 TUP through Dialogic software. The TUP Virtual CIC feature meets all standard Dialogic software requirements, including fault tolerance and SS7 hot-swappable redundancy. This means the TUP Virtual CIC does not affect connectivity or performance.

Please note the following:

- A maximum of 2,048 Virtual CICs can be configured in one SS7 card.
- Product Licensing is supported for TUP Virtual CICs.

Software Architecture

The TUP Virtual CIC protocol controls Circuit Identification Codes (CICs), which function as virtual entities that are not physically attached to the CSP. Management of the physical voice circuits is given instead to the host, and the CSP manages only the signaling links, which are the single physical connection to the SS7 network. The Matrix Controller Series 3 is used only for SS7 card configuration, and for passing messages between the host and the SS7 card.

This message handling bypasses all CSP Call Control processing and functionality. Messaging for call processing is done by *PPL Event Indication* and *PPL Event Request* messages only.

Related Documents

Please refer to the following recommendation for software requirements:

- ITU-T Q.721 - Q.724 (Blue Book TUP)
- *API Reference*

Modules

Dialogic's SS7 protocol software supports Telephone User Part (TUP) call control, which is the basis for the TUP Virtual CIC. Most TUP Virtual CIC messages map directly to host API messages.

Dialogic uses the PPL environment to utilize the TUP Virtual CIC; this environment consists of the following modules, also shown in the figure below for virtual CIC software architecture:

- CSP Signaling, including Layer 3 Plus (L3P) and Layer 3 Plus TUP (L3P TUP), which is modified to support Virtual CICs.
- TUP
- Message Transfer Part (MTP), including MTP2, and MTP3

Each module contains one or more PPL components, which are automatically used when a module is selected. The PPL components are described in *SS7 PPL Information*.

The figures below illustrate the functional modules involved in the TUP Virtual CIC implementation of Virtual CICs on the SS7 card, and show how the signaling stack integrates into CSP architecture.

Figure 5-1 Virtual CIC Software Architecture

.

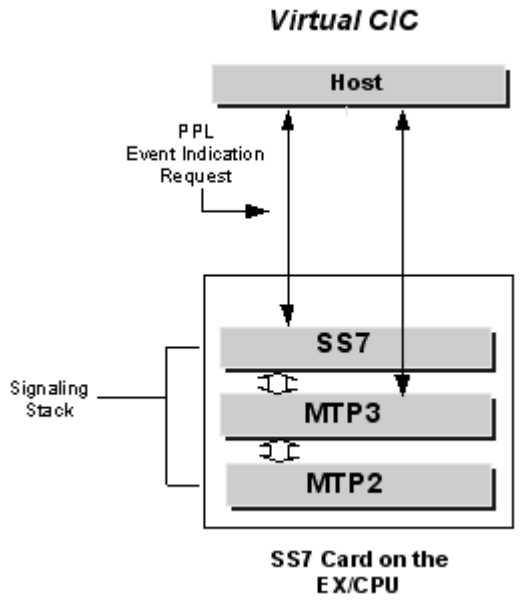
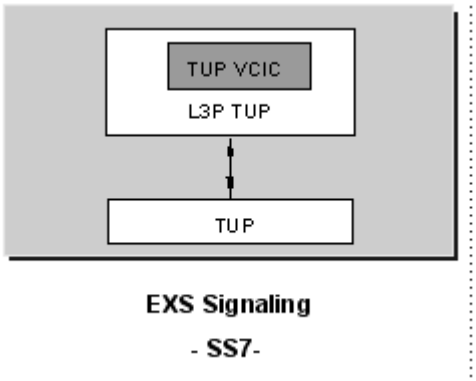


Figure 5-2 L3P TUP with TUP Virtual CIC



MTP Message Tracing in TUP Virtual CIC

The principal objective of the SS7 MTP Message Tracing feature is to send the Raw Data between MTP3 and MTP2 to the host. This will occur in response to the *PPL Configure* message with a “Diagnostic Enabled” option, which allows all MTP Raw Data (both transmitted and received) to be presented to the host. If that option is disabled, then MTP3 will stop sending Raw Data to the host.

TUP Virtual CIC Call Control Management and Maintenance

Overview

The host manages the TUP Virtual CIC call control and circuit management. The Virtual CIC sends all call processing messages directly to Host. Virtual CICs are different from L3P TUP CIC:

Host and CSP Interaction

- The host is notified of service state changes and channel purges by the *PPL Event Indication* message, rather than by the DSO Status Change message.
- PPL event indications and requests are sent for an IAM instead of generating an outseize or receiving an RFS. In the event of call failure, there is no Outseize Control NACK and a *PPL Event Indication* with the reason for the call failure is sent to the host.
- The *Outseize Control* message is not used for L3P Virtual CICs.
- PPL event indications and requests are sent for Alerting (ACM), Answer (ANx) and Release (CLF).
- The host is notified when L3P CIC receives an incoming Continuity Test (COT) message request and the host then performs the loopback, without any CSP Call Control interaction.
- The host reports the span status to the CSP using the *PPL Event Request* message. L3P CIC takes appropriate action.
- If L3P CIC performs a purge, CSP Call Control is not informed. Instead, a *PPL Event Indication* for a CIC status change is sent to the host. This *PPL Event Indication* is equivalent to a *DSO Status Change* message for a physical CIC. It contains an ICB of subtype 0x22 with data length of two:

  Data [0] -- Status (1=OOS, 2=INS)

  Data [1] -- Status change reason (such as purge code)

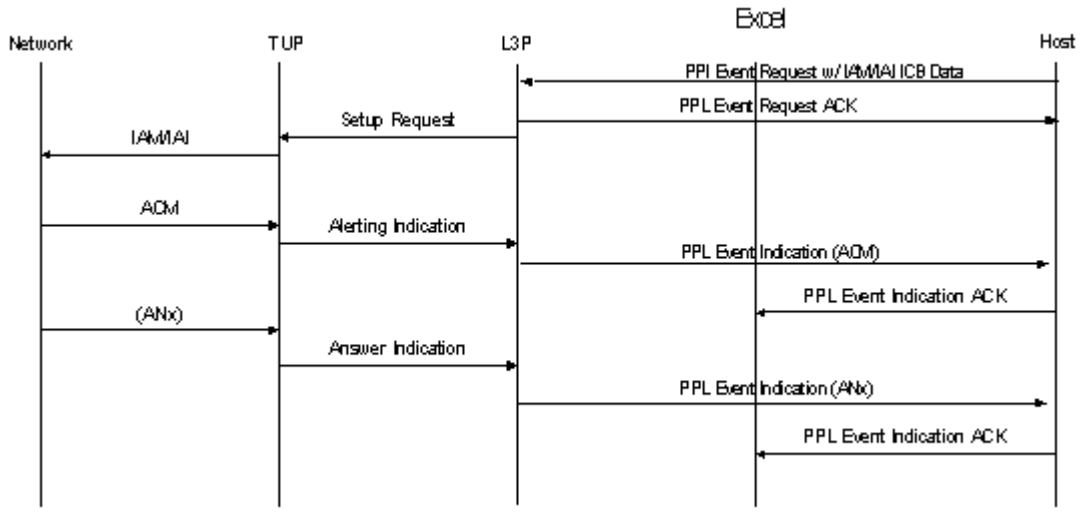
  When we initiate a purge the channel goes to Out of Service and comes In Service immediately. Thus, when we purge OOS, we send the purge Reason in Data [1]. If Out of Service occurs for any other reason, Data [1] will be “0”, as shown in the following table:

Data [1] OOS (0x0199)	Data [1] INS (0x0196)
00 = Host/Normal	00
01 - FF = Purge Reasons	

Example Outgoing Call Flows

This section includes call flow examples for the TUP Virtual CIC outgoing call scenarios for ITU.

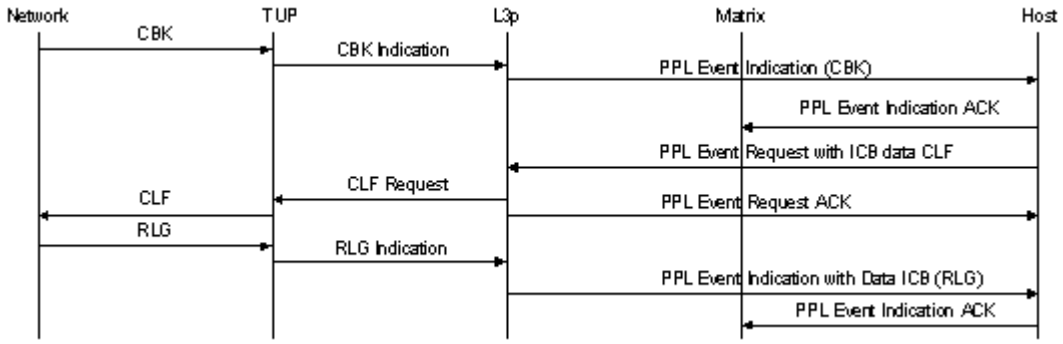
Outgoing Call



1. The host sends to L3P a *PPL Event Request* message, which contains an SS7 Parameters ICB with IAM or IAI ICB data.
2. L3P sends an acknowledgment back to the host and generates a Setup Request to TUP.
3. TUP will generate an IAM or IAI request to the network, depending on the ICB data. Any protocol timers there are will be started.
4. When the network receives the IAM or IAI message, it sends an ACM to TUP, which in turn sends an Alert Indication to L3P. Then L3P sends a *PPL Event Indication* ACM to the host. This means that when the host receives the ACM, it has the complete address and does not need additional digits. The host acknowledges the *PPL Event Indication* to the CSP.

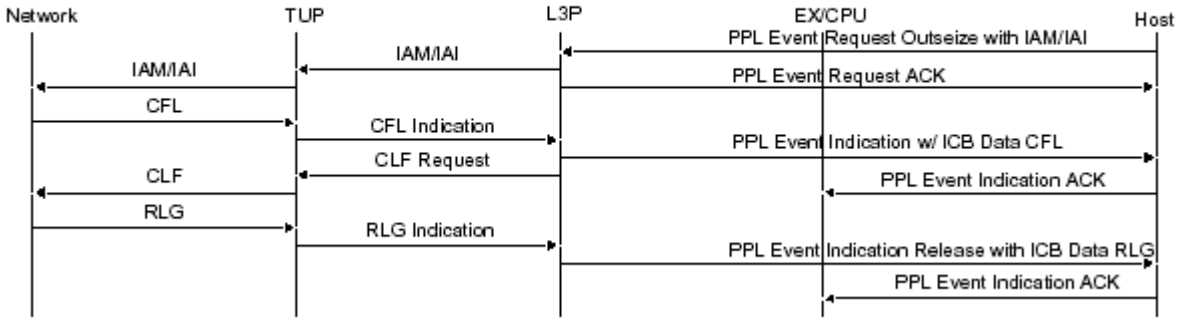
5. The network sends an ANx (indicating the type of call) to TUP, which sends an Answer Indication to L3P.
6. L3P generates a *PPL Event Indication* containing the ANx to the host. The host sends the acknowledgment to the CSP.

Outgoing Call with Remote Clear Back



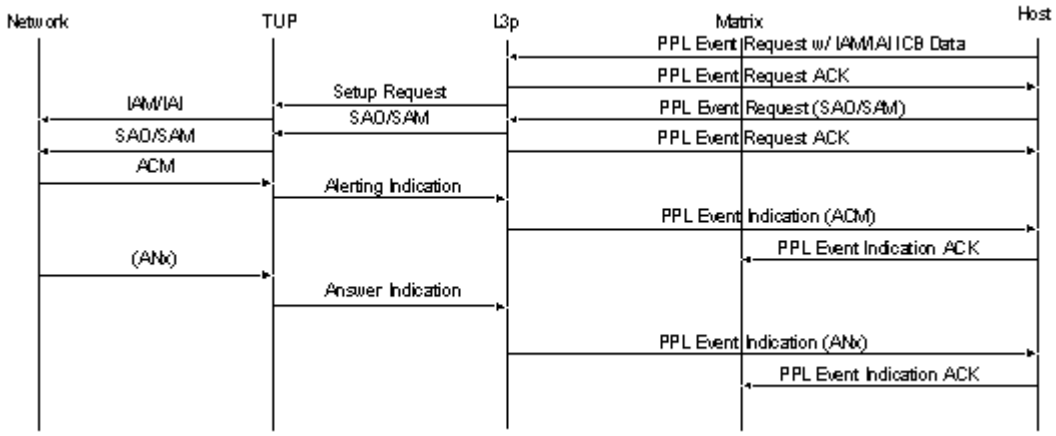
1. The network generates a CBK, clear back message, and sends it to the CSP. TUP sends the message to the host via L3P as a *PPL Event Indication* with CBK.
2. L3P acknowledges the CLF Indication with an RLG request back to the network. The host acknowledges the *PPL Event Indication*. The call is cleared.
3. The host acknowledges receipt of the message and generates a *PPL Event Request*, which contains an SS7 Parameter ICB with parameters for the outgoing CLF message request (for the channel to be released with data). The message is sent to the network.
4. The network responds by generating an RLG acknowledgment. This is sent to the host using the CSP as a *PPL Event Indication* with the Data ICB for RLG. The host acknowledges the *PPL Event Indication*. The outgoing call is released.

Call Failure Received from the Network



1. The host sends an *Outseize Control* message in a *PPL Event Request* to the CSP, which acknowledges the request. The CSP generates an IAM or IAI out to the network for the call.
2. The network sends the host notification of the CFL through the CSP, which generates a *PPL Event Indication* containing an SS7 Parameter ICB, with parameters for the CFL message. This message is sent to the host and acknowledged.
3. The CSP sends the CLF request to the network.
4. The network responds by generating an RLG acknowledgment. This is sent to the host via the CSP as a *PPL Event Indication* with the Data ICB for RLG. The host acknowledges the *PPL Event Indication*. The call is released.

Outgoing Call Sequence

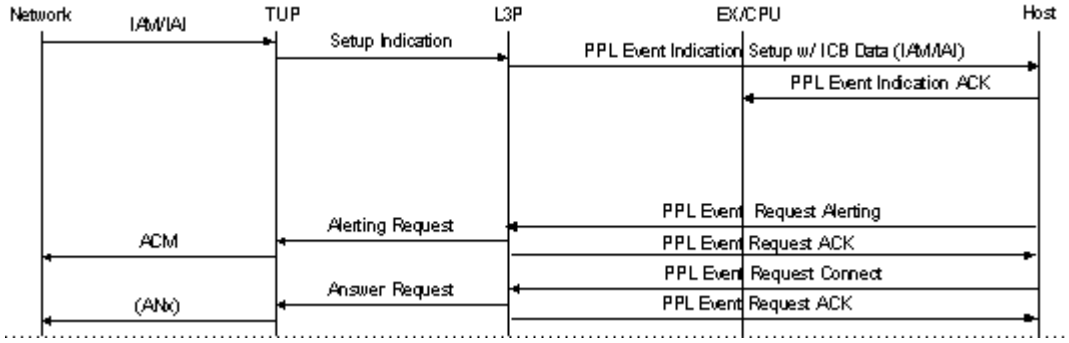


1. The host sends to L3P a *PPL Event Request* message, which contains an SS7 Parameters ICB with IAM or IAI ICB data. L3P sends an acknowledgment back to the host and generates a Setup Request to TUP.
2. TUP will generate an IAM or IAI request to the network, depending on the ICB data. Any protocol timers there are will be started.
3. The host then sends to L3P a *PPL Event Request* message which contains an SS7 Parameters ICB with an SAO or SAM message, indicating the need for additional information (digits).
4. The network receives the IAM or IAI message. It sends an ACM to TUP, which in turn sends an Alert Indication to L3P. Then L3P sends a *PPL Event Indication* ACM to the host. This means that when the host receives the ACM, it has the complete address and does not need additional digits. The host acknowledges the *PPL Event Indication* to the CSP.
5. The network sends an ANx (indicating the type of call) to TUP, which sends an Answer Indication to L3P.
6. L3P generates a *PPL Event Indication* containing the ANx to the host. The host sends the acknowledgment to the CSP.

Example Incoming Call Flows

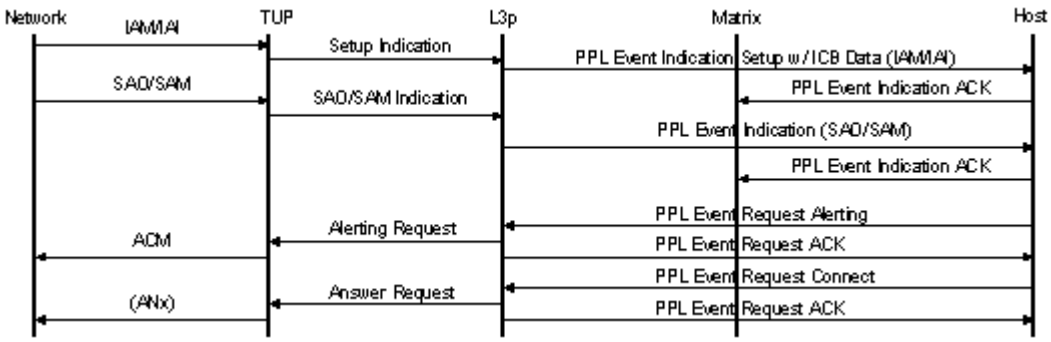
This section includes call flow examples for the TUP Virtual CIC incoming call scenarios for ITU.

Incoming Call



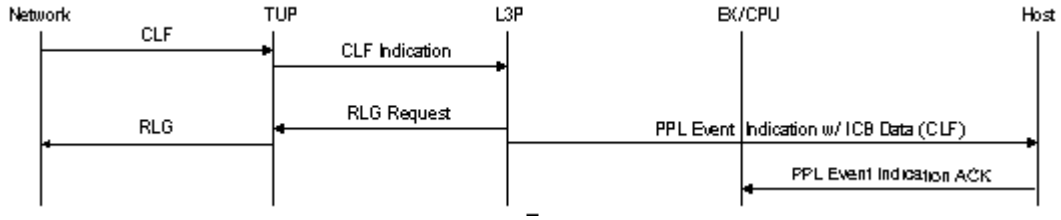
1. L3P alerts the host by sending a *PPL Event Indication* Setup with IAM or IAI ICB Data through the CSP. The host acknowledges the *PPL Event Indication* to the CSP.
2. The host then generates a *PPL Event Request* message to send an Alert Indication to TUP, which sends an ACM to the network. L3P acknowledges the *PPL Event Request*.
3. The host then sends a *PPL Event Request* Connect to L3P, indicating that the call has been answered.
4. L3P acknowledges the *PPL Event Request* and sends an Answer Request message to TUP. Then TUP sends an ANx to the network, successfully completing the procedure.

Basic Call Setup with Overlap Operation



1. The network generates an IAM or IAI message to TUP, which then sends a Setup Indication to L3P.
2. L3P alerts the host by sending a *PPL Event Indication* Setup with IAM or IAI ICB Data through the CSP. The host acknowledges the *PPL Event Indication* to the CSP.
3. If there are not enough digits in the IAM/IAI, the network will send an SAO or SAM to TUP to provide additional information. TUP will generate a *PPL Event Indication* for SAO/SAM and send it to the host. The host acknowledges the *PPL Event Indication* to the CSP.
4. The host then generates a *PPL Event Request* message to send an Alert Indication to TUP, which sends an ACM to the network. L3P acknowledges the *PPL Event Request*.
5. The host then sends a *PPL Event Request* Connect to L3P, indicating that the call has been answered.
6. L3P acknowledges the CLF Indication with an RLG request back to the network. The host acknowledges the *PPL Event Indication*. The call is cleared.

Clear Forward Received from the Network

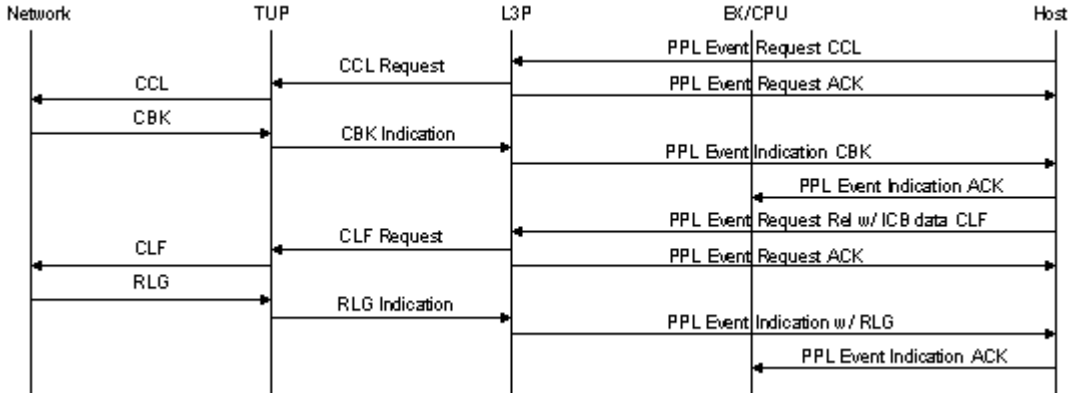


1. The network generates a CLF message clear the call.
2. The message is sent to the CSP, and then to the host using a *PPL Event Indication* message for SS7 ICB Data with CLF.
3. The host acknowledges the CLF message as received.

Example Call Flows

The example call flows included in this section show maintenance and release procedures for incoming and outgoing calls.

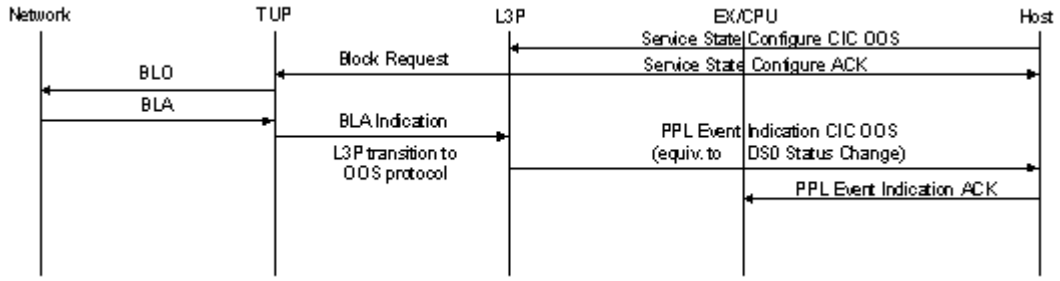
Calling Party Clear



1. The host generates a *PPL Event Request* with the CCL message to clear the call and sends the request to the network via the CSP.
2. The network responds with a Clear Back indication, which is sent in a *PPL Event Indication* from the CSP to the host. The host acknowledges the *PPL Event Indication*.
3. The host then sends a *PPL Event Request* which contains an SS7 Parameter ICB with parameters for the outgoing CLF message request (for the channel to be released with data). The message is sent to the network.

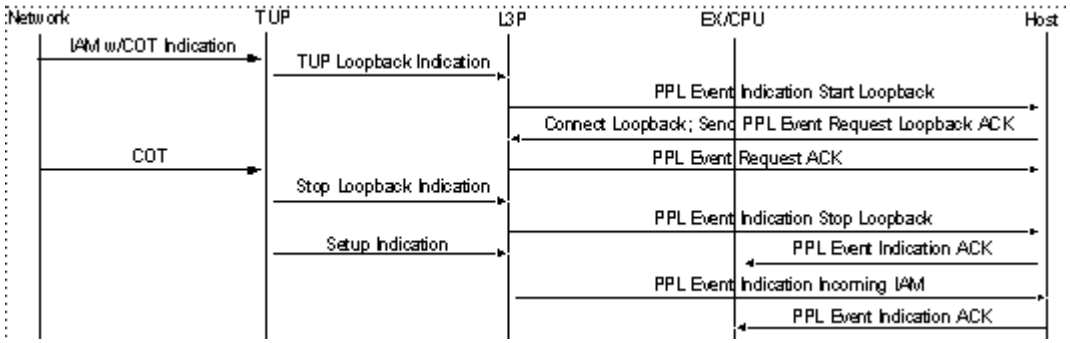
4. The network responds by generating an RLG acknowledgment. This is sent to the host via the CSP as a *PPL Event Indication* with the Data ICB for RLG. The host acknowledges the *PPL Event Indication*. The outgoing call is released.

Channel OOS Sequence



1. The host sends a *Service State Configure* to put the CIC Out of Service.
2. The CSP sends a BLO (default OOS operation) to the network.
3. The network sends a BLA to the CSP, which sends a *PPL Event Indication* for CIC OOS to the host. The CIC is put OOS and marked unequipped after receiving BLA from the network. This *PPL Event Indication* message is the same as sending a DSO Status Change.

COT Incoming Sequence

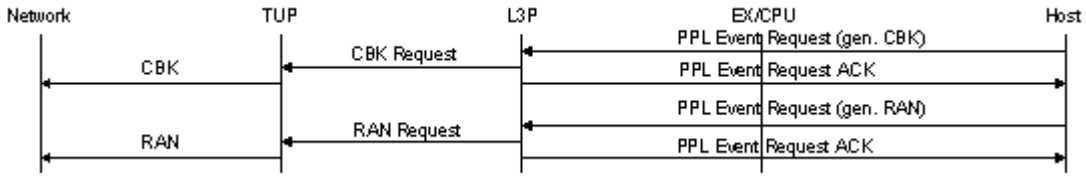


1. When an IAM is received with a continuity test indication, L3PTUP will send a *PPL Event Indication* to the host. The host is controlling the physical circuits; thus, the host will connect the

loopback and send a loopback acknowledgment to the CSP 2000, so that it can synchronize the CIC state with the physical CIC state.

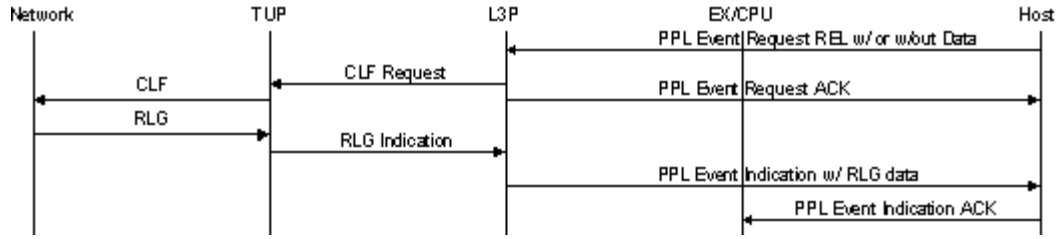
2. After the continuity test is successful, COT will be received from the network and the CSP 2000 will send a *PPL Event Indication* to the host to stop the loopback. The host should remove the loopback.
3. L3PTUP will receive a setup indication from TUP. L3PTUP will send an Incoming Call *PPL Event Indication* to the host.

Clearback and Re-answer



1. The host generates a *PPL Event Request* with the CBK message and sends the request to the network via the CSP. L3P sends acknowledgment to the host.
2. The host then sends a *PPL Event Request* which contains an SS7 Parameter ICB with parameters for the outgoing RAN message request (a re-answer signal for recorded announcements). The message is sent to the network.

Outgoing Call Release, Send Clear Forward



1. The host acknowledges the *PPL Event Indication* and the outgoing call is released.

Incoming Call Trace

PPL event request IAM on stack0, cic 0100

H->X [00 21 00 44 00 00 ff 00 01 31 07 00 00 00 09 87 01 00 00 11 00 3c 01 02 1c 09 01 01 01 03 04 87 26 23 91]

X->H [00 07 00 44 00 00 ff 00 10]

PPL event indication incoming call on stack1, cic 0100

X->H [00 28 00 43 00 7f ff 00 01 31 07 01 00 00 01 23 01 00 00 11 01 91 01 02 1c 10 01 01 03 01 01 0a 02 02 00 00 03 04 87 26 23 91]

PPL event request ACM on stack1, cic 0100

H->X [00 1e 00 44 00 00 ff 00 01 31 07 01 00 00 01 23 01 00 00 11 00 01 01 02 1c 06 04 01 01 0c 01 05]

X->H [00 07 00 44 00 00 ff 00 10]

PPL event indication ACM on stack0, cic 0100

X->H [00 1e 00 43 00 80 ff 00 01 31 07 00 00 00 09 87 01 00 00 11 00 2e 01 02 1C 06 04 01 01 0c 01 05]

PPL event request answer on stack1, cic 0100

H->X [00 1b 00 44 00 00 ff 00 01 31 07 01 00 00 01 23 01 00 00 11 00 05 01 02 1c 03 06 01 00]

X->H [00 07 00 44 00 00 ff 00 10]

PPL event indication answer on stack0, cic 0100

X->H [00 1b 00 43 00 86 ff 00 01 31 07 00 00 00 09 87 01 00 00 11 00 2f 01 02 1C 03 06 01 00]

PPL event request release on stack0, cic 0100

H->X [00 15 00 44 00 00 ff 00 01 31 07 00 00 00 09 87 01 00 00 11 00 3d 00]

X->H [00 07 00 44 00 00 ff 00 10]

PPL event indication clear forward on stack1, cic 0100

X->H [00 1b 00 43 00 87 ff 00 01 31 07 01 00 00 01 23 01 00 00 11 01 97 01 02 1C 03 06 04 00]

PPL event indication release guard on stack0, cic 0100

X->H [00 15 00 43 00 83 ff 00 01 31 07 00 00 00 09 87 01 00 00 11 01 9b 00]

Configuring Virtual CIC Format

Purpose

This section describes how to configure the format for TUP Virtual CIC.

Before you begin

You should be aware that physical CICs can be configured along with Virtual CICs on the same stack.

Form of Addressing

For TUP Virtual CIC, all API messages use the DPC/Stack/CIC form of addressing instead of time slot addressing in the CSP. *SS7 CIC Configure* and *Service State Configure* messages are used. When configuration is finished, the TUP Virtual CIC can generate outbound calls and receive inbound calls, using the *PPL Event Indication* and the *PPL Event Request* messages to perform all call processing functions.

As with the Dialogic China TUP, the TUP Virtual CIC also supports User-Defined messages.

Configuration

This procedure describes how to configure TUP Virtual CIC in addition to the general CSP configuration.

1 Configure the following in the order listed below:

SS7 Cards - Use the CCS Redundancy Configure message.

Signaling Stacks - Use the SS7 Signaling Stack Configure message.

Signaling Link Sets - Use the SS7 Signaling Link Set Configure message.

Signaling Links - Use the SS7 Signaling Link Configure message.

Signaling Route(s) - Use the SS7 Signaling Route Configure.

2 Assign Virtual Voice Circuits. (Virtual CICs). Use the SS7 CIC Configure message.

3 Bring spans and signaling links in-service. Use the Service State Configure message.

4 Bring Virtual CICs in-service. Use the Service State Configure message with new AIB Address Type.

5 Synchronize actual physical CIC states with Virtual CIC states. Use the L3P TUP PPL (0x11) Event Request 64 for Virtual Span INS status.

6 Assign and configure spans. Use the Assign Logical Span ID and T1/E1 Span Configure messages.

7 Configure SS7 PQ cards. Use the CCS Redundancy Configure message.

8 Configure Signaling Stacks. Use the SS7 Signaling Stack Configure message.

9 Configure Signaling Link Sets. Use the SS7 Signaling Link Set Configure message.

10 Configure Signaling Links. Use the SS7 Signaling Link Configure message.

11 Assign SS7 Virtual Voice Circuits (Virtual CICs). See *More on Configuring SS7 Virtual CICs (5-34).*

12 Bring spans and signaling links into service. When all the configuration is complete, use the *Service State Configure* message to establish a connection with the network and to begin call processing. When the destination becomes accessible, the CSP sends the host a *PPL Event*

Indication message. The associated signaling link must be in service for all CICs.

> ***Error Condition***: If a host link failure occurs, all virtual CICs that have been configured in the system will be placed Out of Service. The host must then send a *Service State Configure* message to bring all of those virtual CICs into service.

13 Bring Virtual CICs into service. Send *PPL Event Request 64* and *Service State Configure* to bring Virtual CICs into service. The new AIB Address Type for Virtual CICs is included with both the *Service State Configure* message and the *PPL Event Request 64*. When the Virtual CICs come in-service, a *PPL Event Indication* with Status INS message will be sent for each CIC.

END OF STEPS

More on Configuring SS7 Virtual CICs

All configuration of Virtual CICs is performed using the *SS7 CIC Configure* message. Call control, circuit supervision, and maintenance can be managed using the Telephone User Part (TUP) interface.

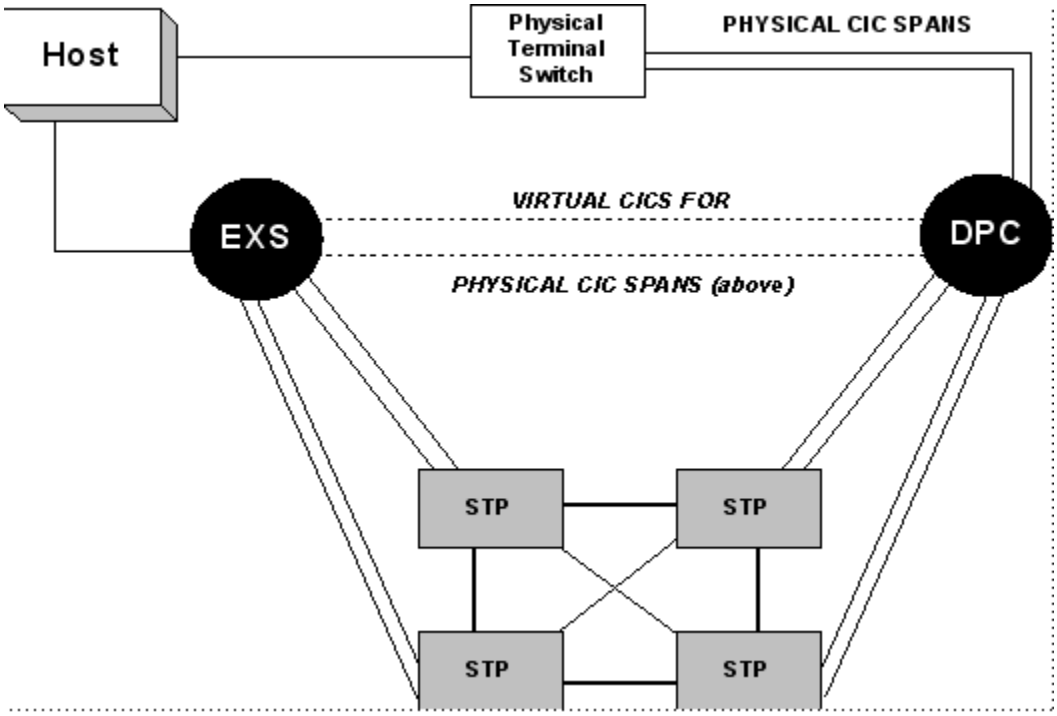
Assigning CICs to Virtual Voice Circuits

Voice circuits controlled by SS7 Signaling Links are identified by a CIC. A specific voice circuit is identified in the network by its unique DPC-CIC combination.

The *SS7 CIC Configure* message assigns a DPC/CIC code pair and a User Part (TUP) to virtual voice circuits.

The *SS7 CIC Query* message is used to retrieve configuration information.

Figure 5-3 Assigning Virtual CICs



Example Trace

The following example shows the *SS7 CIC Configure* message used to configure a Virtual CIC Group. Each CIC Group can have up to 31 circuits for E1 and 24 circuits for T1.

Byte 23 configures the Call Control User Part as TUP Virtual CIC (0x81).

```
Trace: H->X FE 00 15 00 6A 00 SN FF 00 01 14 07 00 01 00
   00 18 00 01 02 03 01
```

BYTE	Field	Value
0	Frame	0xFE
1-2	Length (MSB, LSB)	0x00, 0x16 (22)
3-4	Message Type (MSB, LSB)	0x00, 0x6A
5	Reserved	0x00
6	Sequence Number	0xSN
7	Logical Node ID	0xFF
8	AIB Address Method	0x00

BYTE	Field	Value
9	Number of Address Elements	0x01
10	Type	0x14
11	Data length	0x07
12	Data [0] Stack ID	0x00 (Stack 0)
13-14	Data [1-2] Base CIC Number	0x01, 0x00
15-16	Data [3-4] Base CIC Span	0x00, 0x00 Reserved (should use "0")
17	Data [5] Base CIC Channel	0x00 Reserved (should use "0")
18	Data [6] Number of CICs in Group	0x18 (24 CICs)
19-22	DPC	0x00, 01, 02, 03
23	Call Control User Part	0x81 (TUP Virtual CIC)
24	Checksum	CS

Important! Base span number and base channel should not be used and should be "0".

Matrix Controller Switchover after Configuration

During a matrix controller switchover, PPL Event Indications that come from the SS7 card will be lost. Since all the call processing information is sent to the host using PPL Event Indications, the host will lose some call processing data.

After a matrix controller switchover, the host will get *PPL Event Indication 400* (0x190) from PPL Component SPRC, containing stack ID, DPC, Base CIC and status for all virtual CICs.

SS7 Redundancy

The TUP Virtual CIC supports redundancy for Virtual CICs in the same way that it is currently supported for physical (actual) CICs. There is no change in the configuration of SS7 redundancy.

System Configuration Message

The *System Configuration* message allows you to enable and to configure various system maintenance and monitoring features, including:

- Host Link Failure Detection
- System Busy

Host Link Failure Detection and Virtual CICs

The Host Link Failure Detection feature brings all channels or all spans out of service if the CSP detects a failure in the host link. The CSP determines the integrity of the host link by monitoring host ACKs to *Poll* messages. (You must enable polling to use the Host Link Failure Detection feature.) If the host does not acknowledge two consecutive *Poll* messages, the CSP begins a timer. If the CSP does not receive a message from the host before expiration of the timer, it resets its Ethernet Receiver.

Upon detection of a host link failure, the CSP responds according to the option set by the host, either to bring all channels out-of-service (0x00) or to bring all channels and spans out-of-service (0x01). The *System Configuration* message is used to configure both the Response to Host Link Failure (Data[1]) and the Failure Confirmation Timer (Data[2,3]).

During host link failure, all virtual CICs configured in the CSP will be put Out of Service (OOS). The host will have to send a *Service State Configure* message to bring all those CICs back into service, once the host link failed condition has been corrected.

System Busy Condition and Virtual CICs

A System Busy condition occurs when the CSP is overloaded due to a high volume of call processing traffic. A System Busy alarm is sent to the host when the threshold is reached, indicating that no incoming or outgoing calls can be handled until the condition clears. Although subsequent messages from the host are not necessarily lost, Dialogic recommends that the host hold back on call processing until the System Busy clears.

During a System Busy condition, the matrix controller will send a message to all the SS7 stacks configured on various SS7 cards to inform about System Busy or System Busy Clear; the appropriate signaling procedure will be initiated to notify the network about the CSP condition. No new incoming calls will be processed during a System Busy condition.

This functionality allows the right SEC message to be sent to the network in response to an incoming call, enabling that call to be cleared.

Introduction to BT IUP

Overview

The British Telecom Interconnect User Part (BT IUP) feature offered by Dialogic provides the signaling procedures, formats, and codes that are used to support standard customer services and network features at the point of interconnect between public networks in the United Kingdom using the C7 IUP protocol. The BT IUP feature was derived from the standards of the Public Network Operators - Interconnect Signaling Committee (PNO-ISC).

The common call control procedures are drawn upon as necessary to implement the various services provided in the BT IUP protocol. Dialogic provides BT IUP as a variant of the TUP and L3P TUP modules of an SS7 protocol stack. Once configured, the host may issue messages to place outgoing calls and receive incoming calls.

Purpose

This section describes the use of the BT IUP protocol, provides an example of a configuration file used to set up an SS7 protocol stack for BT-IUP and shows call flows for the BT IUP.

Message Processing

The Dialogic implementation of the BT IUP protocol allows the host to issue any IUP message embedded with the BT IUP Parameters data ICB (0x23). This ICB is formatted from multiple Tag Length Value (TLV) elements. One TLV element that will always be present contains the “H0” and “H1” bytes. These bytes are listed in the H0/H1 Allocation Code tables of the PNO-ISC specifications. Additional TLV elements are used to specify any missing parameters in the underlying IUP message. The L3P BT IUP PPL component contains control logic and atomic functions to retrieve and validate TLVs as required, based upon the initial derived message type. That is, derived from the H0 H1 TLV. L3P BT IUP supplies default values to complete a message so that the message can be sent to a far-end CSP.

To further support off loading the host of routine message processing, the BT IUP Parameters data ICB may appear in Outseize Control messages. When the BT IUP Parameters data ICB is present in the message new activity may be undertaken within the L3P BT IUP PPL component on behalf of the host. The L3P BT IUP PPL has access to an atomic function which tests for the presence of this ICB and CSPs to user-desired PPL logic based upon the actual message type.

Important! This implementation is coded to the CSP for "standard" processing. Any user-desired processing may be added as desired.

Features Available

In addition to the basic features the host application may include:

- Answer an incoming call, stimulate the host to place an outgoing call, and then connect both.
- Place an outgoing call by forming the smallest message possible.
- Place an outgoing call utilizing the BT IUP message of the host's design.
- Automate the typical host call processing by off-loading routine operations to the PPL (L3P IUP PPL).
- Remove functionality from either of the PPLs and allow the host to perform as much of the protocol as desired.
- Allow Dialogic customers to receive and transmit every BT IUP message currently specified, or to be specified.
- Support Call Failure Reporting to the host for both incoming and outgoing calls.
- Support Circuit Group Out of Service
- Support common Querying Messaging
- Support Audit Logging
- Support Error Alarms
- Support Fault Logs
- Support redundancy

Messages Not Supported

The following messages are not supported. These messages may be passed from the host to the network via PPL Event Requests, and may be passed from the network to the host in PPL Event Indications. Therefore, any processing required on their behalf are handled entirely by the host. All other messages listed in PNO-ISC/SPEC/006 Issue 3 May 1999 are supported.

- Call Drop Back (CDB)
- Enveloped ISUP Message (EIM)
- Enveloped ISUP Segmented Message (EISM)
- Nodal End-to-END Data (NEED)
- Operator Condition Message (OCM)
- Node-to-Node Message (OSS)

- Protocol Negotiation Message (PNM)
- User-to-User Data (UUD)

Congestion Control

Congestion control is supported for BT IUP. See *Enabling Congestion Control for BT IUP (5-55)* for more details.

Configuring BT IUP on an SS7 Card

Purpose This section describes how to configure BT IUP on an SS7 card.

Before you begin Acquire the Originating Point Code (OPC) and Adjacent Point Code (APC) before configuring the SS7 stack.

Configuring BT IUP The steps below are used for an SS7 card.

Step	Action	API Message(s)
1	Assign and configure spans	Assign Logical Span ID T1/E1 Span Configure
2	Configure SS7 cards	CCS Redundancy Configure
3	Configure Signaling Stacks	SS7 Signaling Stack Configure
4	Configure Signaling Link Sets	SS7 Signaling Link Set Configure
5	Configure Signaling Links	SS7 Signaling Link Configure
6	Configure Signaling Route(s)	SS7 Signaling Route Configure
7	Assign Voice Circuits (CICs)	SS7 CIC Configure
8	Bring spans, CICs and signaling links in-service	Service State Configure

Configuration Sequence Use the following configuration sequence to bring SS7 signaling links and voice circuits in service.

1. Assign and configure spans.

Assign Logical Span IDs to spans on T1, E1, or J1 line cards you want to use for SS7 with the *Assign Logical Span ID* message.

Set the span format to Clear Channel by using the *T1 Span Configure* or *E1 Span Configure* messages. This step prevents line cards from attempting to extract CAS signaling from the spans.

2. Configure SS7 cards.

Configure SS7 cards in the system as primary or secondary using the *CCS Redundancy Configure* message. Independent cards are assigned as a primary card. In a redundant pair of cards, one is assigned as the primary card and the other as the secondary.

Subsequent SS7 configuration messages are sent only to primary SS7 cards, which forward the configuration to the secondary card if there is one.

- When configured as a pair, SS7 cards operate in a primary/ secondary mode with respect to the call control user part. Signaling links operate in a load-sharing mode.
- Configuration information can be retrieved with the *CCS Redundancy Query* message.

See *Configuring SS7 Card Redundancy (2-30)* for more information.

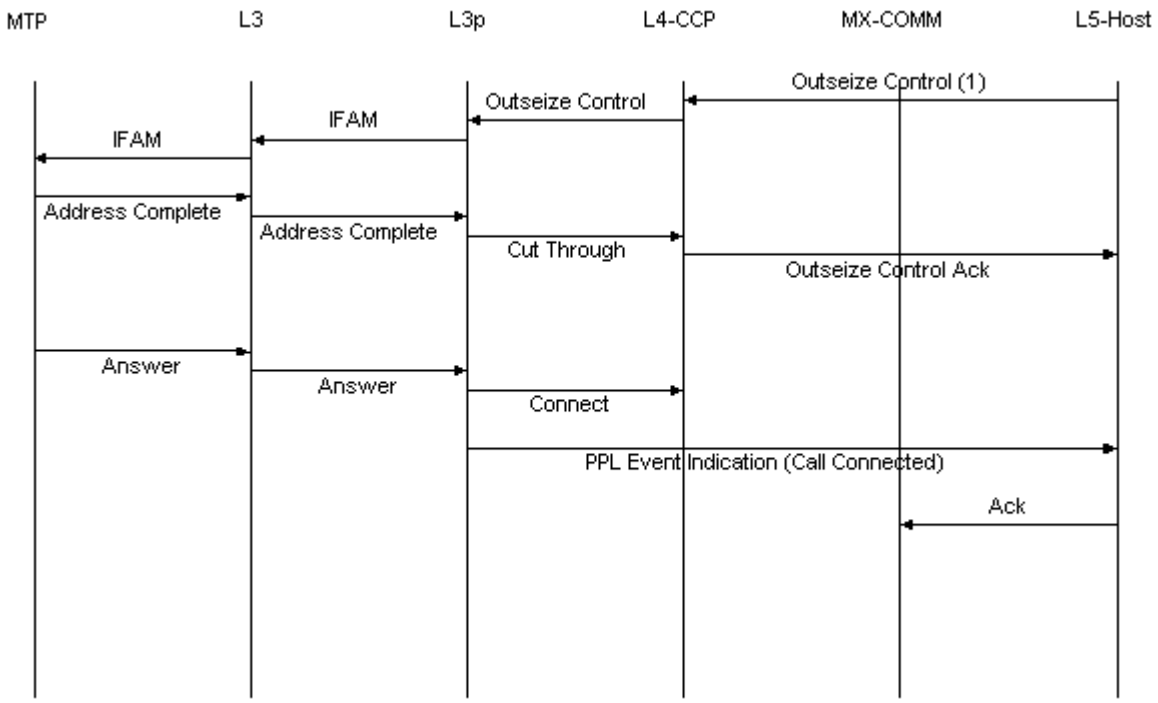
BT IUP Call Flows

Purpose This section includes BT IUP outgoing and incoming call flows.

FAM The *PPL Event Request* (SND) and *PPL Event Indication* (FAM/SAM) sequence may continue until a FAM is returned to the Host or the Host halts sending of *PPL Event Requests* (SND).

Successful Outgoing IFAM Call

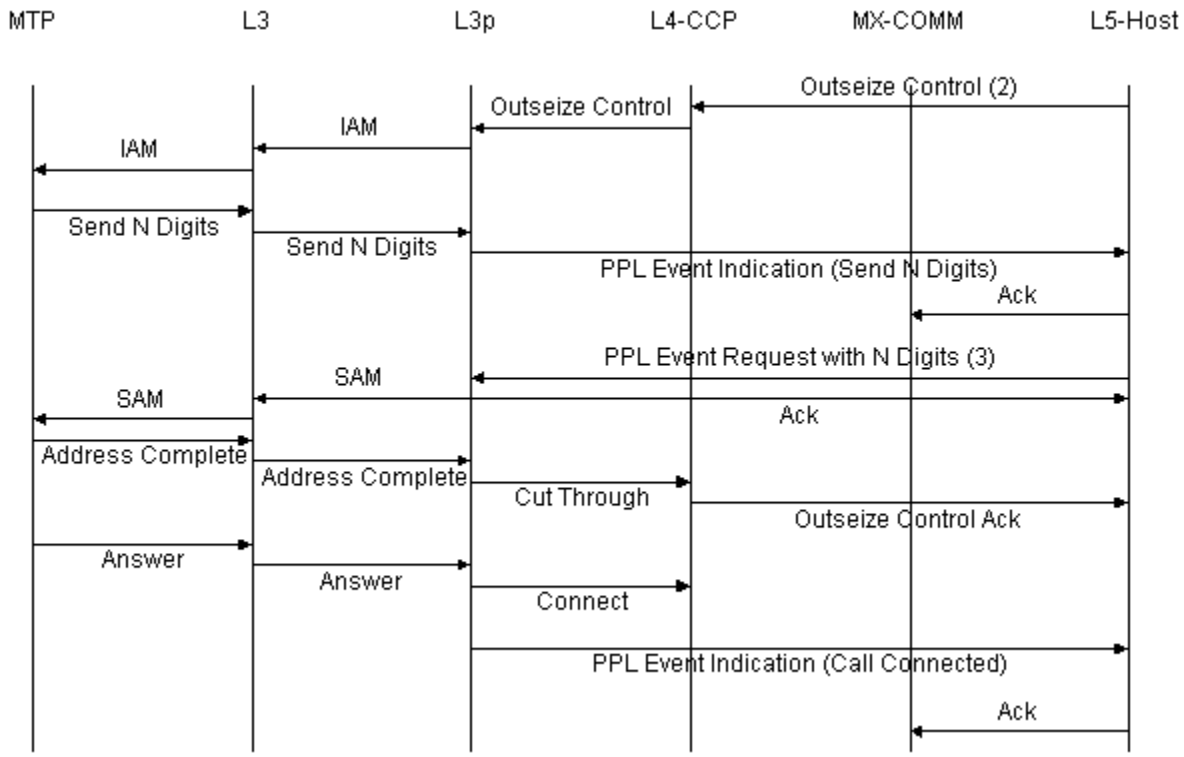
The following is a call flow diagram in which a successful call is originated from the host. All address digits are present in the Outseize Control message, therefore an IFAM is issued to the lower layers.



(1) The Outseize Control Message may be composed using either the IUP ICB or the Outpulse Stage N Address Data ICB and the Call Type ICB. When the IUP ICB is presented it is sent to lower layers unmodified. This example assumes that the HOST has built an IFAM message within the IUP ICB. When the Outpulse Stage N Address Data ICB format is chosen one or two BCD strings may be encoded in the ICB. When only one BCD string is encoded it will be interpreted as the called address. When two BCD strings are encoded the first one is

interpreted as the calling address and the second as the called address. All other information required to format an IFAM will be extracted from the Config Bytes.

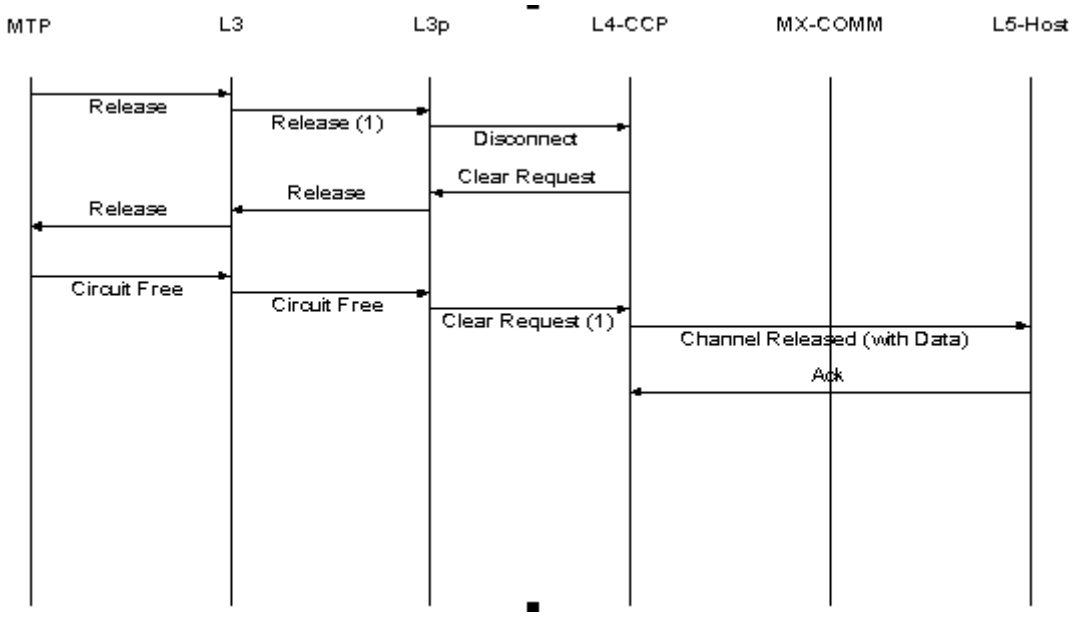
Successful Outgoing IAM Call Host Supplies Subsequent Digits



(2)The only way to cause an IAM is to encode an Outseize Control Message with a IUP ICB containing an IAM.

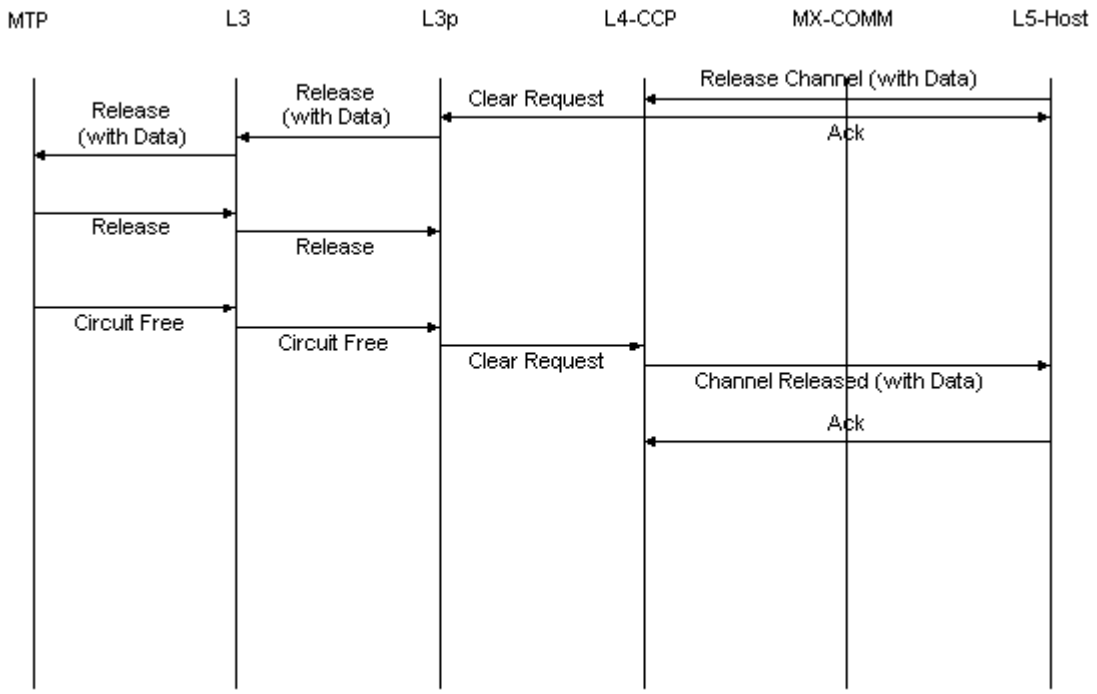
(3)This message may contain either a IUP ICB with the exact SAM or a generic Outpulse Stage N Address ICB.

Called Party Clears Call



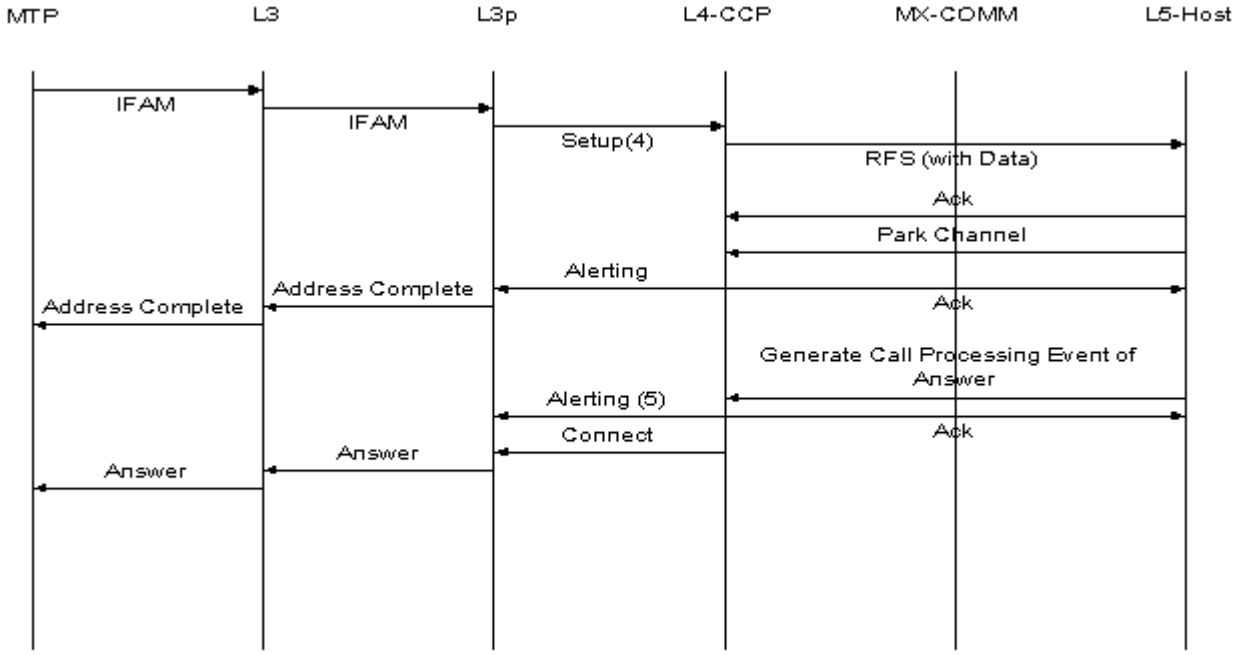
(1) The Release reason is saved by L3P and is forwarded in the Clear Request to L4-CCP.

Calling Party Clears Call



Successful Incoming Call for an LSU

This example assumes that the CSP is the Local Switching Unit (LSU) (that is, terminating switching functionality), since they only receive IFAM messages. When configured as an Intermediate Node (switching A Digital Main Switching Unit (DMSU)) IAM may begin the message exchange.

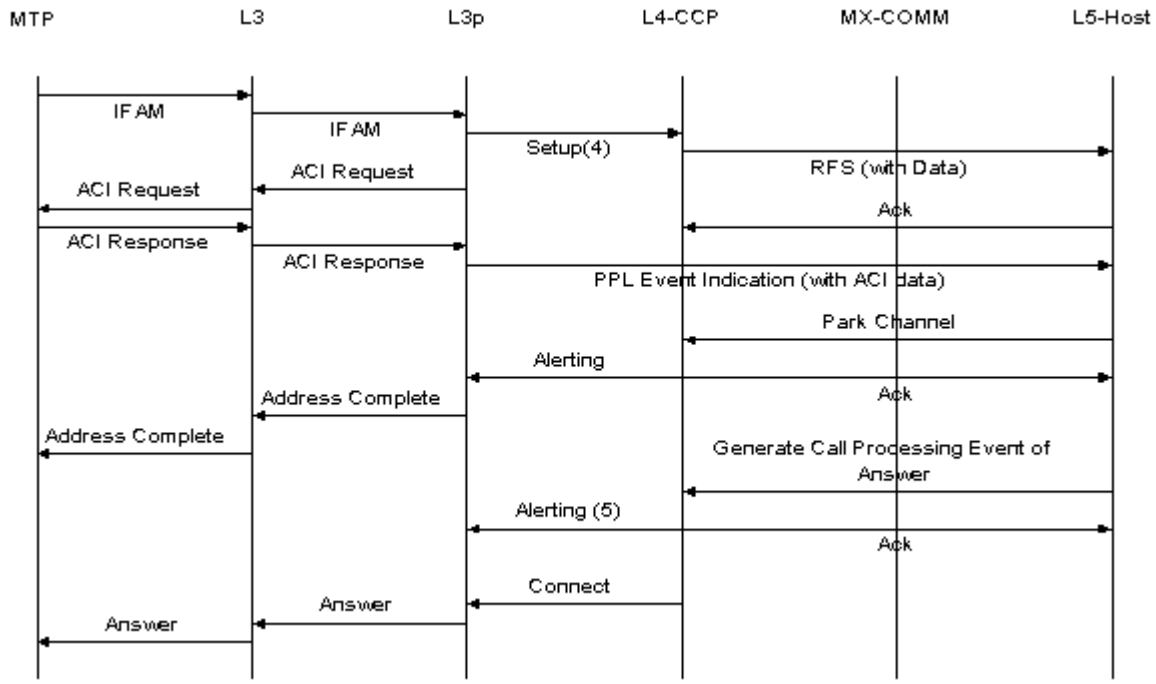


This message may contain either the received IUP ICB (i.e. containing the complete IFAM) from the IFAM or a BCD-Encoded Stages Data ICB. Which ICB type to attach to the Setup message is dependent upon a Config Byte defined above.

This message will be ignored by L3P IUP.

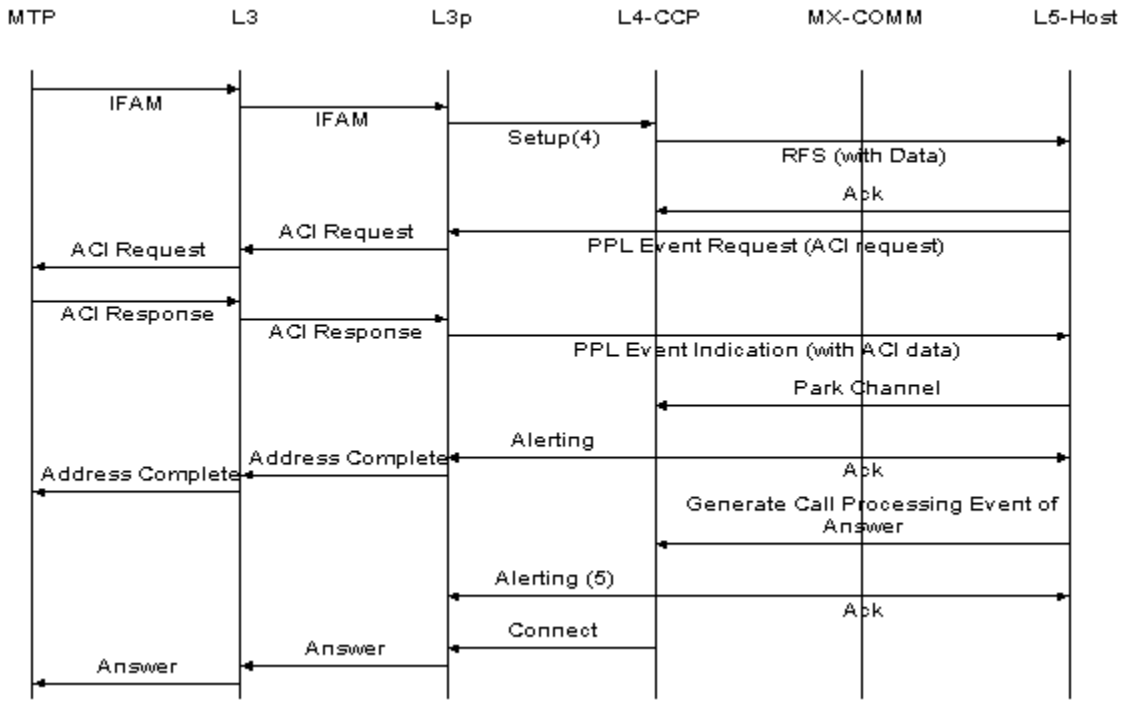
Incoming Call - Simple Telephone Call - Auto ACI Generation

This call flow depicts a situation where a Config Byte has been set to allow the automatic generation, via L3P, of Additional Call Information Requests (ACI) to the calling network.



Incoming Call - Simple Telephone Call - Host Control with ACI

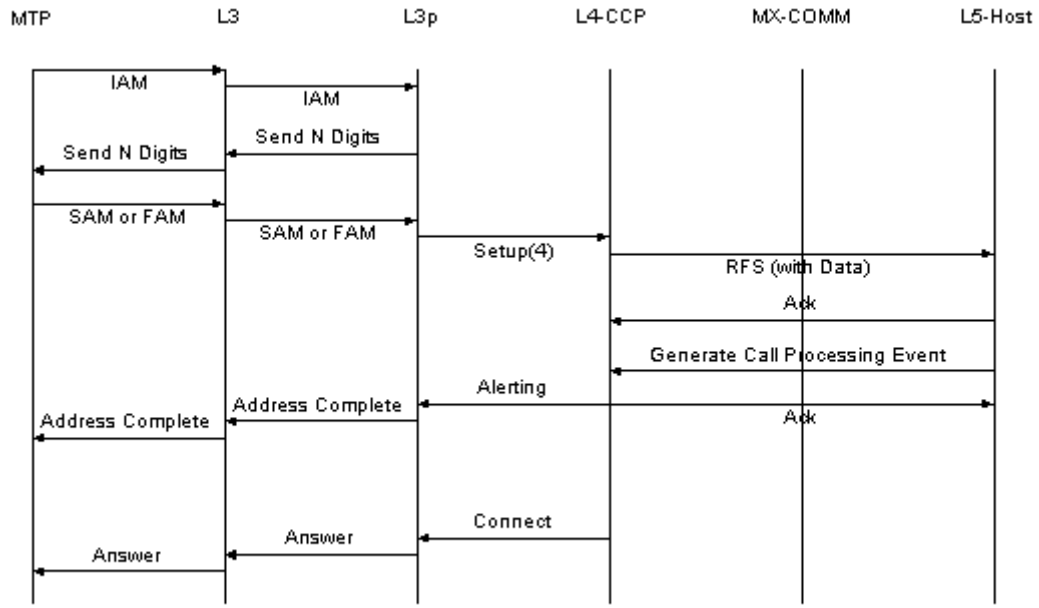
This call flow depicts a situation where a config byte has been set to allow the host to control the requests for ACI data.



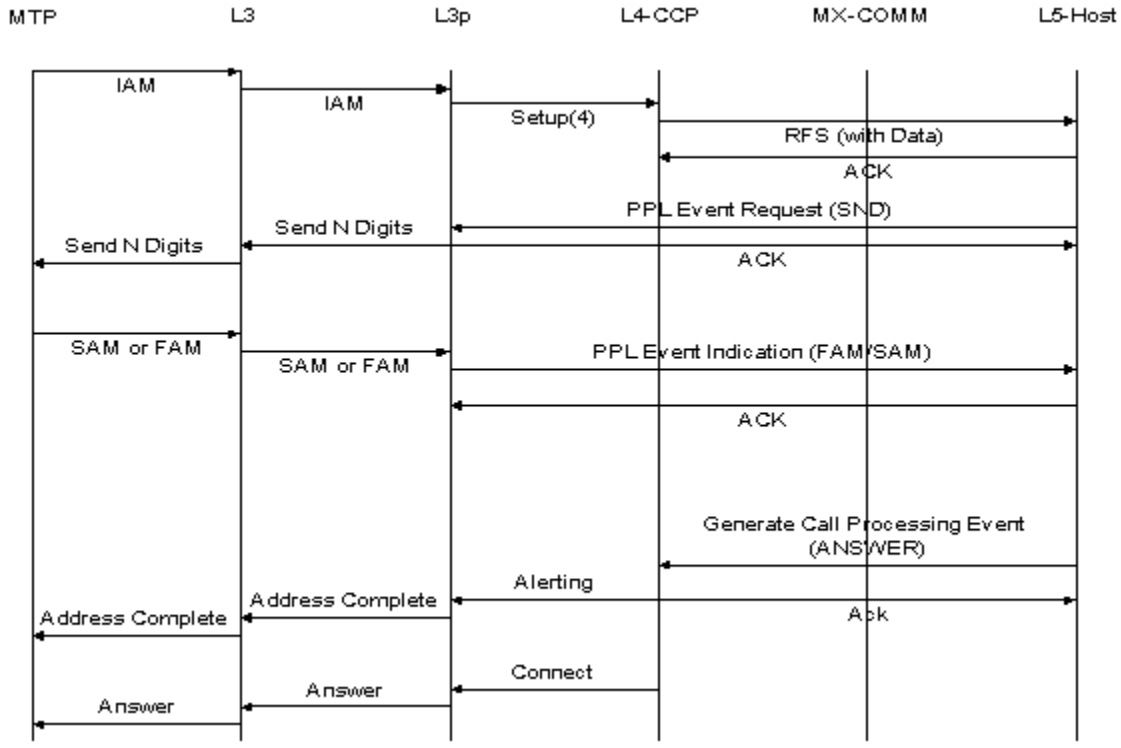
Successful Incoming Call Flows

For a DMSU where L3P IUP completes IAM Processing (AUTO SND)

A Digital Main Switching Unit (DMSU) must be able to process IAM and IFAM messages since this type of CSP may appear between two LSUs. The more complicated case is the reception of an IAM. L3P IUP analyzes the number of Address digits received. If the required number of digits has not been received within a configurable amount of time, as defined via Config Byte settings, the appropriate number of Send N Digit messages will be issued until the digit count is satisfied.



Successful Incoming Call - Host Control **For an DMSU where L3P IUP completes IAM Processing (Host Control**



BT IUP Configuration Examples

Overview This section provides configuration examples using the BT IUP protocol.

Examples

```
# Assign Logical Spans.
00 0D 00 a8 00 00 01 00 01 11 04 00 00 01 00
00 0D 00 a8 00 00 01 00 01 11 04 00 01 01 01
00 0D 00 a8 00 00 01 00 01 11 04 00 02 01 02
00 0D 00 a8 00 00 01 00 01 11 04 00 03 01 03

# Configure E1 Span format for HDB3, Clear Channel.
00 0F 00 d8 00 00 01 00 01 0C 02 00 00 09 00 F8 D0
00 0F 00 d8 00 00 01 00 01 0C 02 00 01 09 00 F8 D0
00 0F 00 d8 00 00 01 00 01 0C 02 00 02 09 00 F8 D0
00 0F 00 d8 00 00 01 00 01 0C 02 00 03 09 00 F8 D0

# Download SS7 License here.

# Configure CCS Redundancy.
# Primary Slot 8, Secondary Slot 9.
00 0E 00 5B 00 00 01 00 02 01 01 02 01 01 03 00

# Configure the SS7 Stack.
# Stack ID 0, OPC 0x00000123 0-36-3
# 4 Modules
# MTP/ITU, L3P/ITU, L3P-TUP/IUP, TUP/IUP
00 18 00 5c 00 00 01 00 01 21 02 08 00 00 00 01 23 04 01
   01 03 01 04 02 05 02

# Configure the SS7 Linkset
# Linkset 0, APC 0x00000987 1-48-7
00 0F 00 5D 00 00 01 00 01 1E 02 00 00 00 00 00 09 87

# Configure the SS7 Link
# Linkset 0, Link ID 0, SLC 0, Data rate 64k
# Span 0, Channel 30
00 21 00 5E 00 00 01 00 02 1F 03 00 00 00 0d 03 00 00 1e
   00 00 00

# Configure the SS7 Route
# Linkset 0, Destination 0, Route 0, DPC 0x00000987 1-48-
   7
00 13 00 5F 00 00 01 00 01 20 05 00 00 00 00 00 00 00 09
   87 00 00 00 FF FF

$cic:                    (0100)
$log_span_id:            (00 02)
```

```
$channel:               (00)
$nbr_cics:              (01)
$call_ctrl_user_part:   (01)  ' TUP
00 17 00 6A 00 00 01 00 01 14 07 00 00 00 $log_span_id
   $channel $nbr_cics $dest_point_code
   $call_ctrl_user_part

# SS7 CIC Configure
# These are done span by span. CIC numbering must match on
   each side
# Setup CICs on span w/ Link of SLC=0
00 15 00 6a 00 00 01 00 01 14 07 00    00 00  00 00  00
   0f    00 00 09 87 00
00 15 00 6a 00 00 01 00 01 14 07 00    00 10  00 00  0f
   0f    00 00 09 87 00
'Setup CICs on second span
00 15 00 6a 00 00 01 00 01 14 07 01    00 20  00 01  00
   1f    00 00 09 87 00
'Setup CICs on third span
00 15 00 6a 00 00 01 00 01 14 07 00    00 40  00 02  00
   1f    00 00 09 87 00
'Setup CICs on fourth span
00 15 00 6a 00 00 01 00 01 14 07 01    00 60  00 03  00
   1f    00 00 09 87 00

# Bring Span In Service
00 0D 00 0A 00 00 01 00 01 0C 02 00 00 F0 00
00 0D 00 0A 00 00 01 00 01 0C 02 00 01 F0 00
00 0D 00 0A 00 00 01 00 01 0C 02 00 02 F0 00
00 0D 00 0A 00 00 01 00 01 0C 02 00 03 F0 00

# Bring SS7 Link In Service
00 0D 00 0a 00 00 01 00 01 09 02 00 00 F0 00

# Bring SS7 CIC's In Service
00 13 00 0a 00 00 01 01 02 0D 03 00 00 00 0D 03 00 00 0f
   f0 00
00 13 00 0a 00 00 01 01 02 0D 03 00 00 10 0D 03 00 00 1d
   f0 00
00 13 00 0a 00 00 01 01 02 0D 03 00 01 00 0D 03 00 01 1e
   f0 00
00 13 00 0a 00 00 01 01 02 0D 03 00 02 00 0D 03 00 02 1e
   f0 00
00 13 00 0a 00 00 01 01 02 0D 03 00 03 00 0D 03 00 03 1e
   f0 00
```

BT IUP Signaling Stack Congestion Control

Overview

The Signaling Stack Congestion Control feature prevents the SS7 card from failing due to call traffic overload.

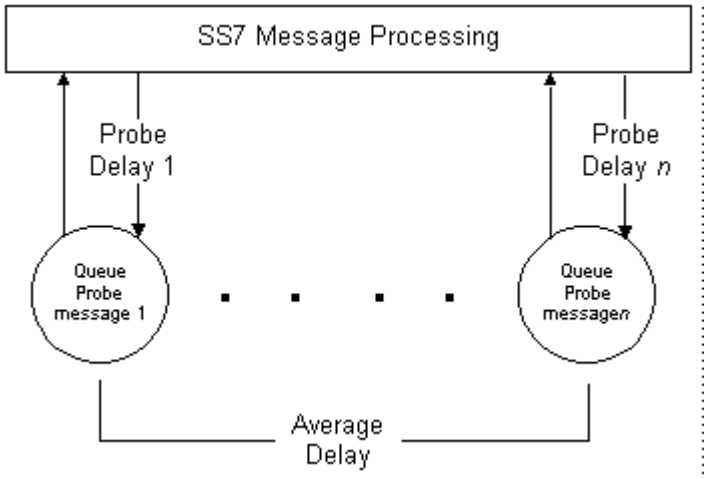
Dialogic's automatic congestion control monitors inbound and outbound traffic (from the CSP perspective). Congestion (at the CSP) is not invoked as a result of an adjacent CSP entering a congested state. When an adjacent CSP enters a congested state, Dialogic-initiated calls to that CSP will fail. A *Release Message* (REL) will be received in the backward direction in response to the *Initial Address Message* (IAM). The cause parameter within the REL message will indicate congestion. It is the host's responsibility to examine the cause parameter in the *Channel Released with Data API*, and throttle (stop) traffic for a host-determined amount of time.

If the CSP encounters congestion in the outbound direction, attempts to *Outseize* will result in a negative acknowledgment of 0x0C (Outbound Congestion). In this case, no IAM/IFAM is sent to the network.

Congestion Levels

The level of congestion on each signaling stack is monitored by the Signaling Procedure Control (SPRC) component. Internal “Queue Probe” messages are periodically sent to measure the amount of delay (backup) in the SS7 message queue. Each time a “Queue Probe” is returned to SPRC, the delay for that probe is recorded, as well as an average delay for ***n*** number of probes. If either delay reaches the configured threshold, a congestion condition is determined to exist and action is taken to relieve the congestion.

Figure 5-4 Calculating Congestion Level



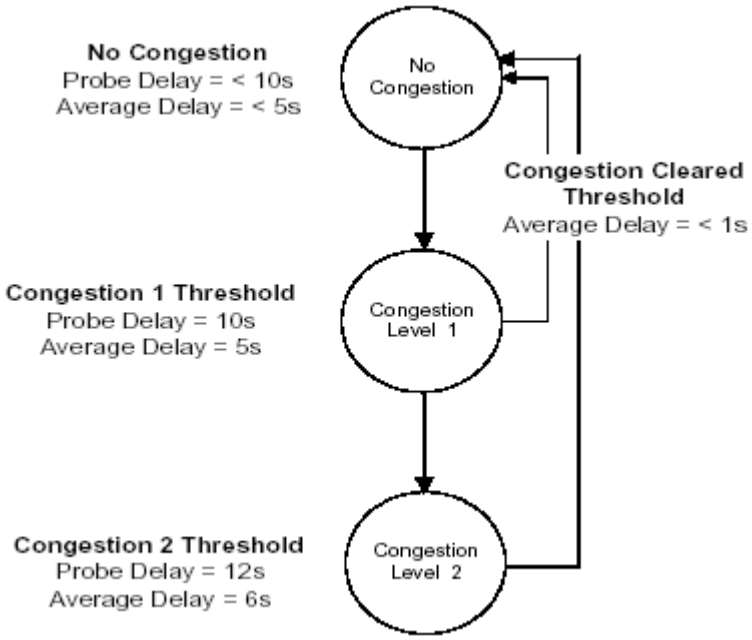
Congestion Status

There are 3 congestion statuses:

- No Congestion
- Level 1
- Level 2

The transition between congestion statuses and the associated thresholds with default settings is shown in the figure below:

Figure 5-5 Congestion Levels



Default Values

The default threshold values are as follows:

- Congestion Level 1
  - Probe Delay: 10 seconds
  - Average Delay: 5 seconds
- Congestion Level 2
  - Probe Delay: 12 seconds
  - Average Delay: 6 seconds
- Congestion Cleared
  - Average Delay: < 1 seconds

Important! The default values for congestion Level 2 are not configurable. The number of probes used to calculate the Average Delay is 10 and the time between probes is one second.

Enabling Congestion Control for BT IUP

Purpose

Congestion Control is enabled for BT IUP by turning on specific configuration bytes that are outlined in this section.

Enabling Congestion Control

To enable the congestion management features you must:

Step	Action
1	Enable L3 BT IUP Config Byte 0x03 for incoming congestion detection.
2	Enable L3P BT IUP Config Byte 0x35 for outgoing congestion detection.
3	Enable Level 2 congestion detection in the MTP3 component 0x2B Config Byte 0x11 by setting the value to 0x01.

Switch Response to Congestion Level 1

If Congestion Level 1 is reached, an *Alarm* message is sent to the host and action is taken on the congested stack.

The *Alarm* message in response to Congestion Level 1 includes the following information:

Severity: Major (0x01)

Entity: General (0x01)

Alarm Number: SS7 Signaling Stack Congestion (0x16)

Data[0] Stack ID

Data[1] Direction

0x00	Incoming
0x01	Outgoing

Data[2] Status

0x01	Congestion Level 1
0x02	Congestion Level 2

The action the CSP takes in response to congestion depends on whether the call is incoming or outgoing.

BT IUP Incoming Calls

If Congestion Level 1 is reached, a CNA message is sent to the call originator. The Return Cause reason is indicated as 0x07, meaning there is congestion.

Outgoing Calls

If Congestion Level 1 is reached, L3P IUP sends an Access Denied indication to the host with a Cause Code of 0x59, which means outseize failed and blocked.

Switch Response to Congestion Level 2

If Congestion Level 2 is reached, all new call MSUs (IAM, IFAM) are discarded by MTP3. The host is sent an *Alarm* message indicating the change in Congestion Status to Level 2 (0x02). All new call MSUs are discarded until the congestion cleared threshold is reached and the stack returns to a No Congestion status.

From congestion Level 2, you can only go to a No Congestion status. The Congestion Status cannot go from Level 2 to Level 1. When the Congestion Cleared threshold is reached and maintained for ***n*** number of probes, the host is sent an *Alarm Cleared* message and normal call processing resumes.

> **Important!** Congestion Level 2 can only be reached for incoming calls. Outgoing call requests are rejected if Congestion Level 1 is reached.

Congestion Cleared

When the congestion condition is reduced below the Congestion Cleared threshold and remains below that level for a specified number of probes, an *Alarm Cleared* message is sent to the host and normal call processing resumes on the stack.

The *Alarm Cleared* message includes the following information:

Severity: Major (0x01)

Entity: General (0x01)

Alarm Number: SS7 Signaling Stack Congestion (0x16)

Data[0] Stack ID

Data[1] Direction

0x00	Incoming
0x01	Outgoing

Data[2] Congestion Status

0x00	Congestion Cleared

Congestion Control Implementation Can be Changed

This section describes modifications you can make to the default Congestion Control implementation.

Congestion levels are monitored by the ISUP SPRC (0x13) or TUP SPRC (0x53) components. To change the default implementation of congestion control, modify the following PPL Config Bytes or Timers as required:

- Congestion Level 1 Probe Delay Threshold - Incoming Calls

  PPL Config Byte 0x29
- Congestion Level 1 Probe Delay Threshold - Outgoing Calls

  PPL Config Byte 0x2D

  See for default values
- Congestion Level 1 Average Delay Threshold - Incoming Calls

  PPL Config Byte 0x2A
- Congestion Level 1 Average Delay Threshold - Outgoing Calls

  PPL Config Byte 0x2E
- Amount of time between probes

  PPL Timer 2
- The number of probes used to calculate the Average Delay

  PPL Config Byte 0x28
- Congestion Cleared Threshold - Incoming Calls

  PPL Config Byte 0x2B
  - Number of probes to determine Incoming Congestion Cleared

  PPL Config Byte 0x2C
- Congestion Cleared Threshold - Outgoing Calls

  PPL Config Byte 0x2F
  - Number of probes to determine Outgoing Congestion Cleared

  PPL Config Byte 0x30

Introduction to SSUTR2

Overview

SSUTR2 provides the facility to carry ISDN access information in the Telephone User Part (TUP) messages. Most of the signaling messages and message formats used in SSUTR2 are the same ones used in TUP. The SSUTR2 interface is transparent to the user and provides basic call setup and circuit maintenance signals, with associated procedures.

Refer to These Specifications

Software requirements are provided in the following recommendation:

- SSUTR2 V11-14T 1995

See also:

- Blue Book TUP Q.721-Q.724
- Signaling, including Layer 3 Plus (L3P) and Layer 3 Plus TUP (L3P TUP), which is modified to support Virtual CICs.

Message Transfer Part (MTP), including MTP2 and MTP3.

Definition

SSUTR2 is variant of the Blue Book TUP and is used to define the signaling interface between a fixed network and a mobile network.

Comparison to TUP

The principal differences between this protocol and TUP exist in the circuit states. SSUTR2 does not support hardware and software group blocking messages; instead, it has defined new circuit states for blocking and for the actions to be taken in those states. Host configuration of these specialized circuits is done using new config bytes. See *SS7 PPL Information*.

- In functionality, SSUTR2 has defined charge messages and procedures for the French National Network.

Modules

Dialogic uses the PPL environment to implement the SSUTR2. The modules shown in Figure 1.1 include:

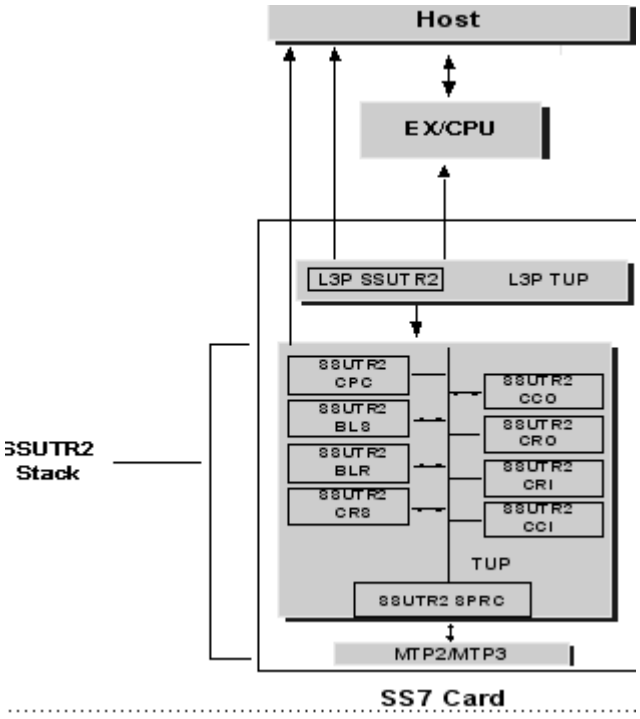
- Signaling, including Layer 3 (LP3) and Layer 3 Plus TUP (L3P TUP), which are modified to support SSUTR2.
- TUP (SSUTR2)
- Message Transfer Part (MTP), including MTP2 and MTP3

Each module contains one or more PPL components, which are automatically used when a module is selected. For the SSUTR2 PPL data, see *SS7 PPL Information*.

Software Architecture

The figure below illustrates the functional modules involved in the implementation of SSUTR2 on the SS7 card, and shows how the signaling stack integrates into the CSP architecture.

Figure 5-6 CSP Architecture



Configuring SSUTR2

Purpose

This section explains how to configure SSUTR2 on an SS7 card and includes information about the PPL components and API messages used with this feature.

Before you begin

The following API messages need be used in the configuration of SSUTR2:

- *Assign Logical Span ID* (0x10)
- *CCS Redundancy Configure* (0x5B)
- *SS7 CIC Configure* (0x6A)
- *SS7 Signaling Link Configure* (0x5E)
- *SS7 Signaling Link Set Configure* (0x5D)
- *SS7 Signaling Route Configure* (0x5F)
- *SS7 Signaling Stack Configure* (0x5C)
- *Service State Configure* (0x0A)
- *T1/E1 Span Configure* (0xA9/0xD8)

Incoming and Outgoing Circuits

SSUTR2 defines specialized incoming and outgoing circuits. Specialized outgoing circuits are available only for outgoing calls, incoming MIFs (Initial Address), continuity recheck request, and blocking request received on these circuits are considered irrational. Similarly specialized incoming circuits are available only for incoming calls, outgoing calls and host-initiated blocking is not allowed on these circuits. A config byte is used in L3PCIC and CPC to allow the host to configure incoming and outgoing specialized circuits.

Configuration

This procedure describes how to configure the SSUTR2 in the general CSP configuration.

1 Assign and configure spans. Use the Assign Logical Span ID and T1/E1 Span Configure messages.

2 Configure SS7 cards. Use the CCS Redundancy Configure.

3 Configure Signaling Stacks. Use the SS7 Signaling Stack Configure.

4 Configure Signaling Link Sets. Use the SS7 Signaling Link Set Configure.

5 Configure Signaling Links. Use the SS7 Signaling Link Configure.

6 Configure Signaling Route(s). Use the SS7 Signaling Route Configure.

7 Assign Voice Circuits (CICs). Use the SS7 CIC Configure.

8 Bring spans, CICs and signaling links in-service. Use the Service State Configure.

END OF STEPS

SSUTR2 PPL Components

For TUP SSUTR2 outgoing continuity, two components are used specifically:

- TCCO--This variant handles outgoing continuity during outgoing call setup.
- SSU CRO--This variant handles the outgoing recheck procedure.

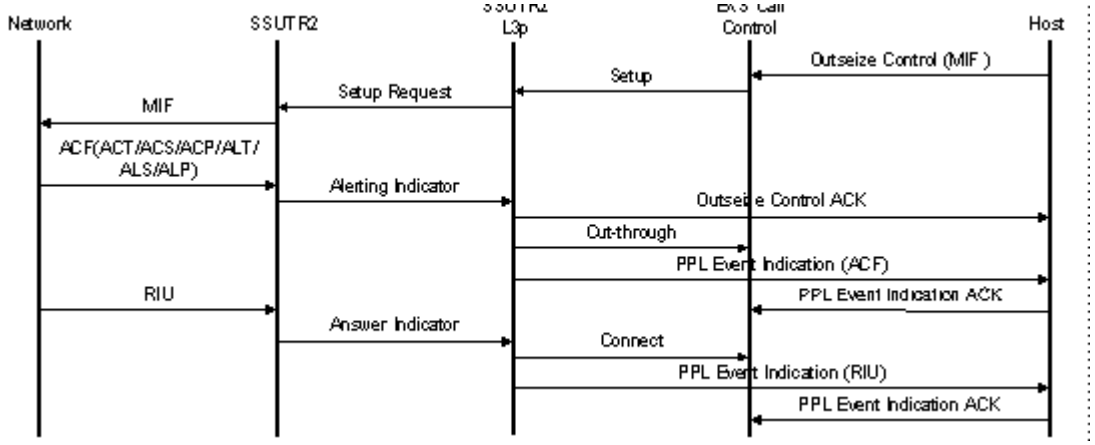
In addition, the PPL Component SCFG is used to configure the SSUTR2 variant and initialize the SSUTR2 PPL components “RW” and “RO”.

SSUTR2 Call Flows

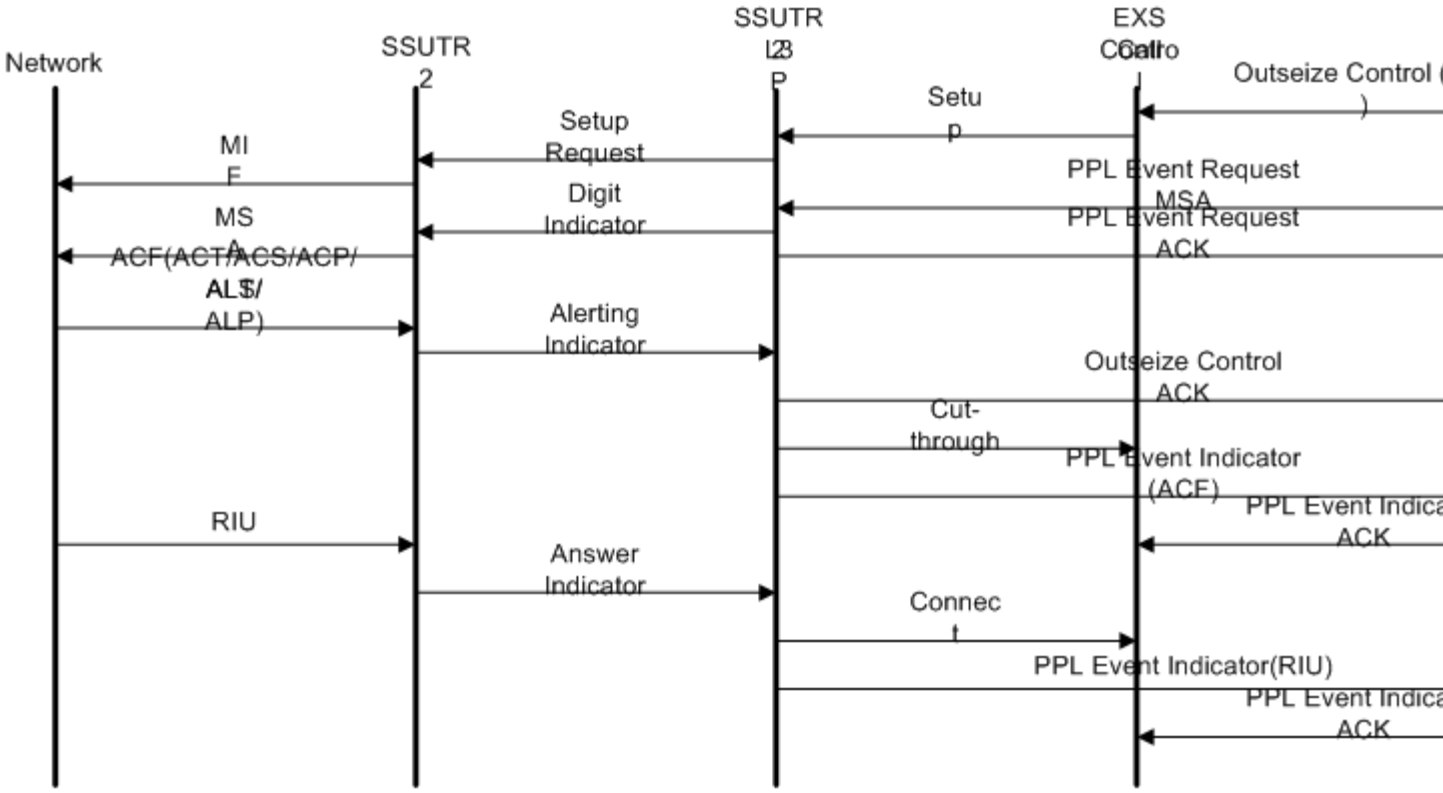
Overview This section includes call flows that demonstrate the use of SSUTR2.

Example Call Flows The following call flows show examples of outgoing and incoming calls.

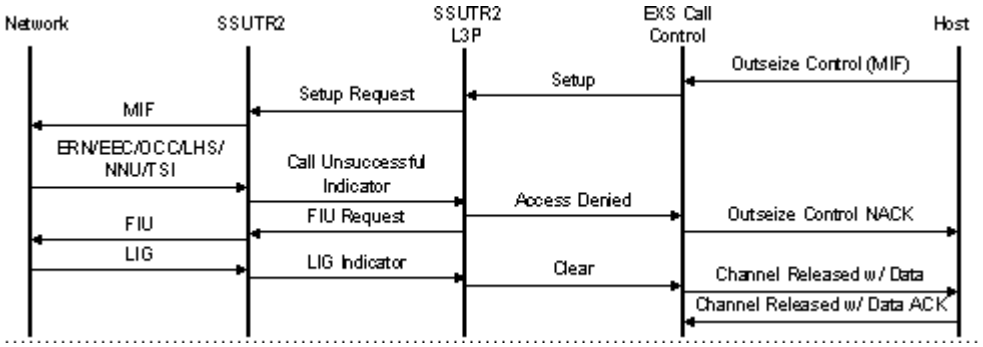
Outgoing Call



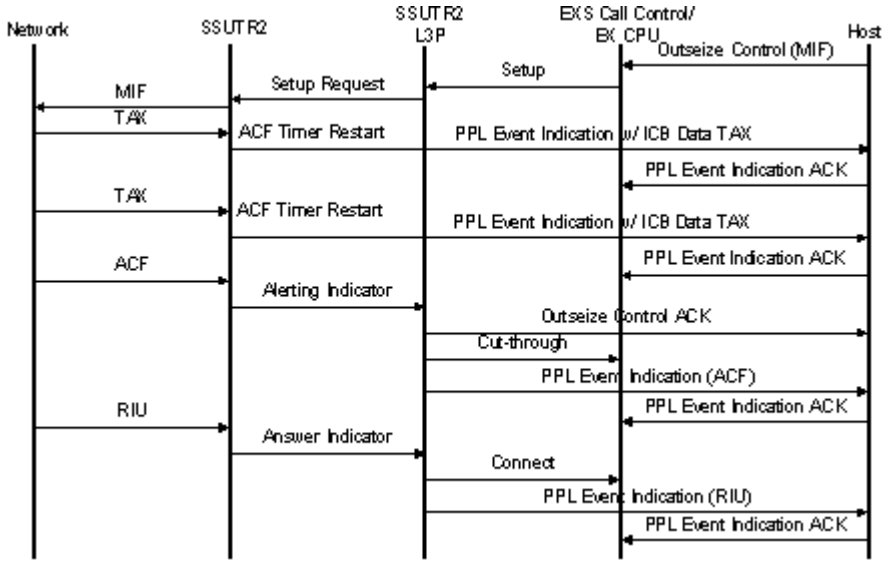
Outgoing Call, Overlap Operation



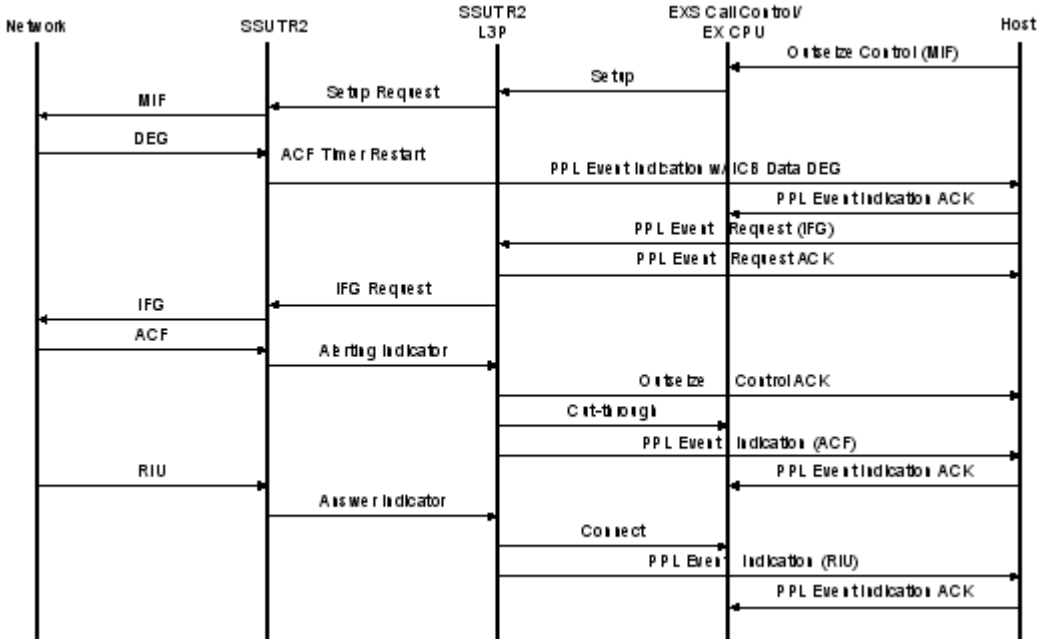
Outgoing Call Failure (Call Failure Signal Handling)



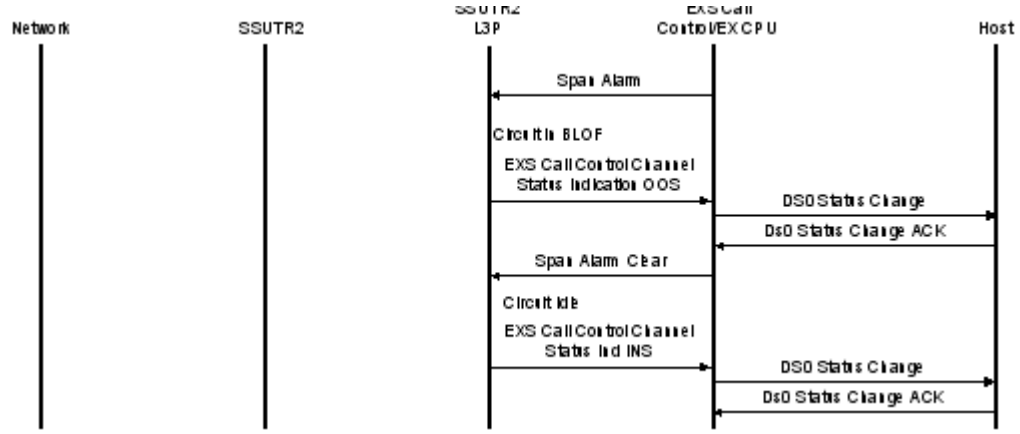
Outgoing Call, Charge Message



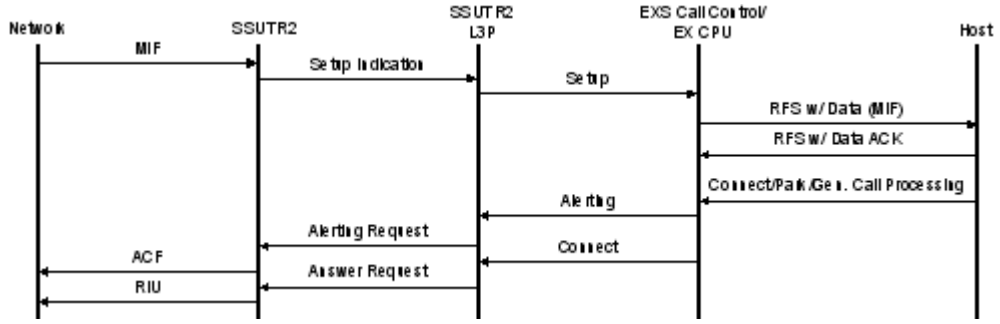
Outgoing Call, Additional Information Requested



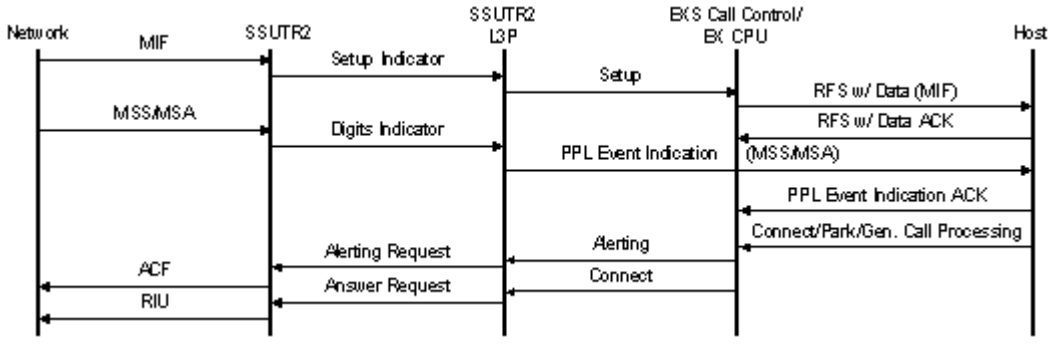
Span Alarm HW Blocking



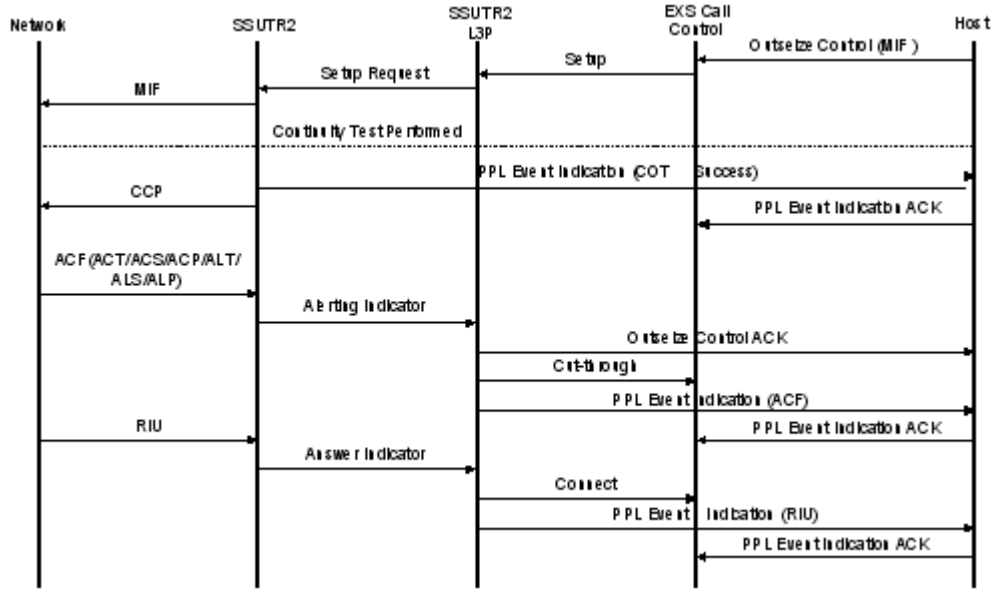
Incoming Call Setup



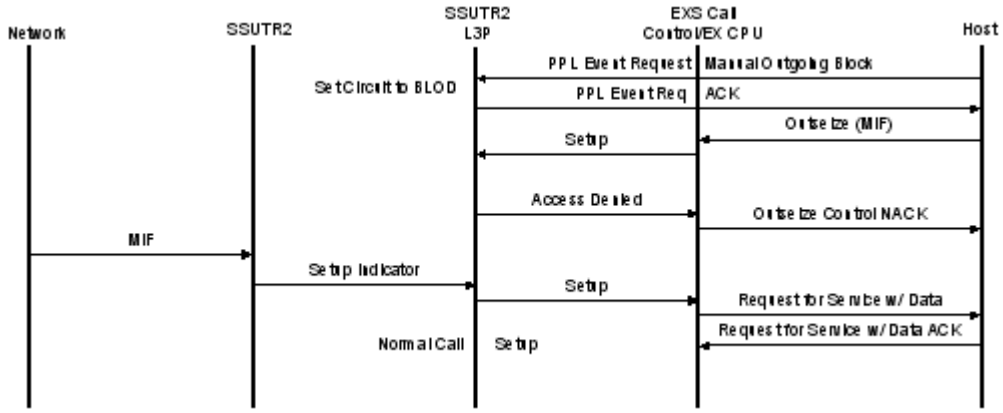
Incoming Call, Overlap Signaling



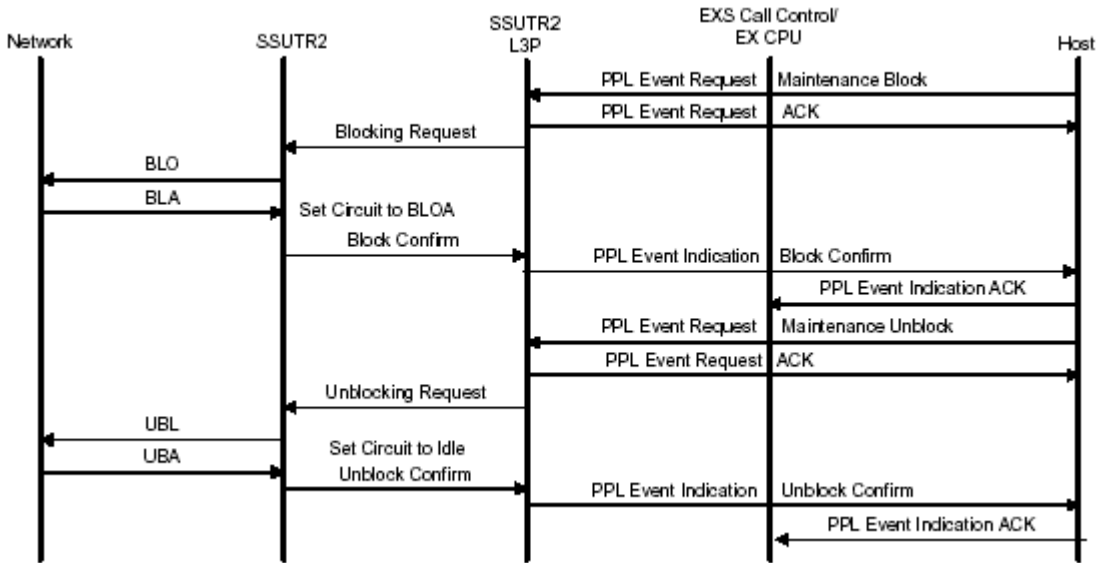
Outgoing Continuity in MIF



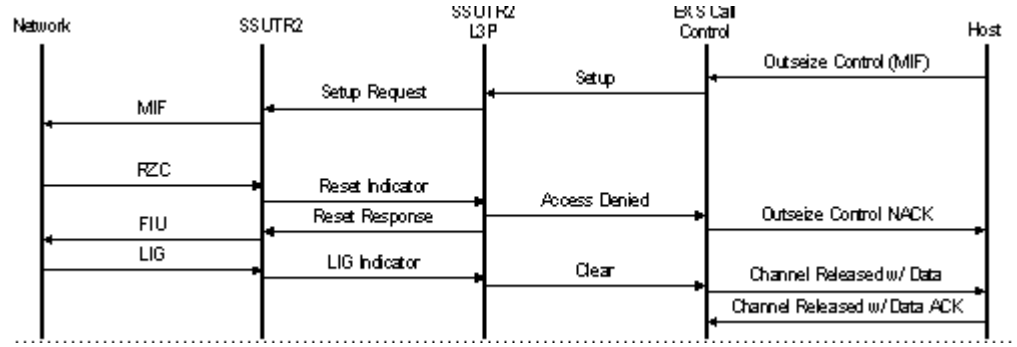
Manual Blocking Outgoing



Maintenance Blocking (Exchange of Blocking Signal)



Reset During Wait for ACF



6 SCCP/TCAP

Purpose The Signaling Connection Control Part (SCCP) and Transaction Capabilities Application Part (TCAP) are parts of the SS7 signaling protocol.

This chapter describes the implementation of the SCCP/TCAP feature in the CSP software.

Information is also provided about configuring SCCP/TCAP options for customization, parameters, segmentation, services, and applications supported.

For information about implementing a TCAP direct host connection on the CSP, see *SCCP/TCAP in the CSP Architecture*.

Introduction to SCCP/TCAP

Overview

SCCP and TCAP support offers advanced routing, data transfer, and management control features. Signaling Connection Control Part (SCCP) provides both connectionless and connection-oriented network services above MTP Level 3. Transaction Capabilities Application Part (TCAP) provides transaction and remote operation capabilities to a large variety of applications distributed over the CSP and service centers in the network. The Dialogic SCCP/TCAP implementation supports the following connectionless services:

- Specialized Routing
- Data Transfer
- Management Control

The SCCP layer and TCAP layer services are directly accessible through the SCCP and TCAP primitives. The option of using SCCP services only or using both SCCP and TCAP services is configurable for each subsystem. This allows users to choose the appropriate protocol interface for customized applications.

SCCP

Together SCCP and MTP are used as the network layer for TCAP-based services. For each component of an SS7 network, SCCP provides a different service to its user which is referred to as the subsystem.

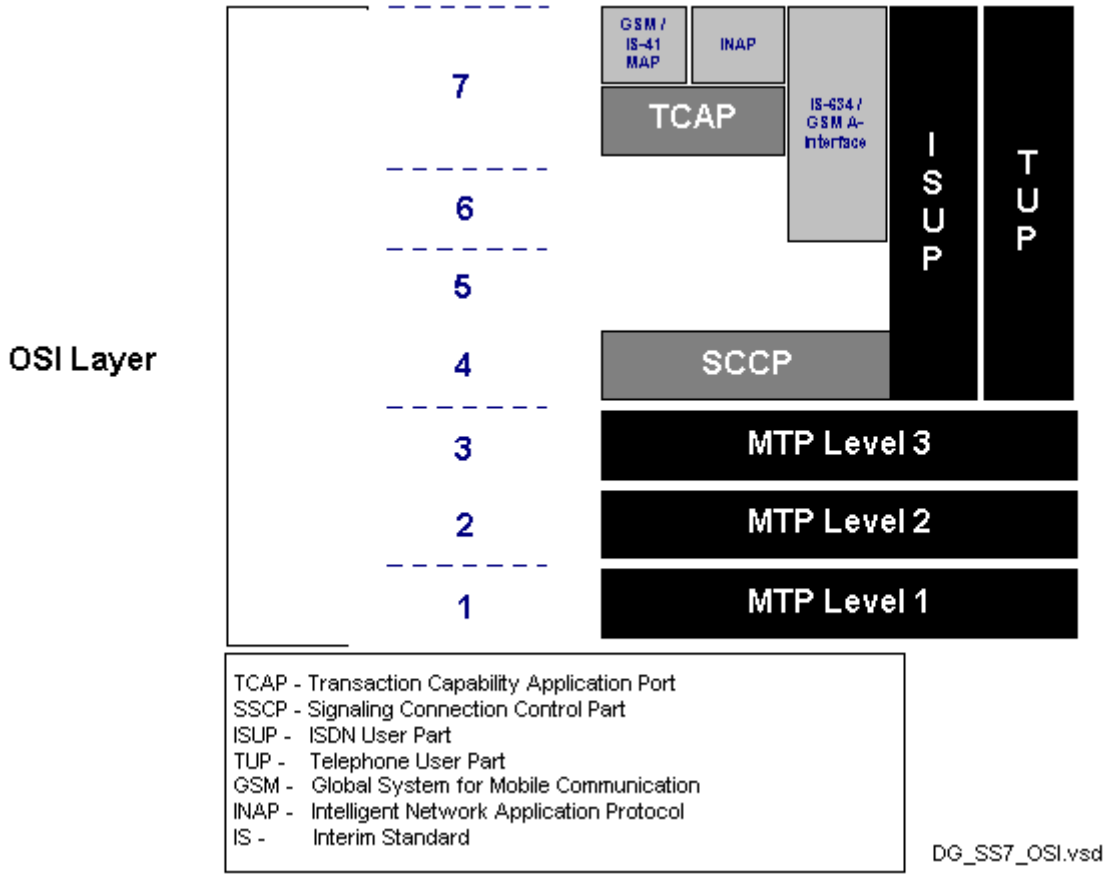
- In a Service Control Point (SCP), the sub-system of the SCCP is a database.
- For an SSP, the sub-system of the SCCP is an application software package.
- For a Signal Transfer Point (STP), the SCCP also provides Global Title Translation.

TCAP

TCAP enables the deployment of advanced intelligent network services by supporting non-circuit related information exchange between signaling points using the SCCP connectionless service. TCAP provides the framework to retrieve information or invoke remote operations.

TCAP offers the means for end users in the SS7 network to query another end office and acts as the software interface between an SS7 office and database services in order to obtain data from the SS7 network.

Figure 6-1 Mapping SS7 stack to OSI layers



SCCP/TCAP in the CSP Architecture

SCCP and TCAP provide the services to the host by a set of SCCP primitives and a set of TCAP primitives which are carried in the *PPL Event Request* and *PPL Event Indication* API messages.

ITU and ASNI/Telecordia Standards - Versions Supported

The following are the standards versions that the SCCP/TCAP implementation supports:

- ITU Q.771-Q.774 TCAP 1997
- ANSI T1.114 TCAP 2000 and Telecordia GR-246 issue 7

- ANSI T1.112 SCCP 2000 and Telecordia GR-246 issue 7

SCCP/TCAP Supports Many Services and Applications

Overview

This section provides examples of the kinds of services and applications that the SCCP/TCAP feature supports.

Assumptions and Dependencies

SCCP software assumes services from SS7 Message Transfer Part (MTP). TCAP software assumes connectionless services from SCCP and services from MTP.

Services enabled by TCAP

The following are some examples of the services enabled by TCAP:

Special Service Number Translation

An SSP uses TCAP to query an SCP to determine the routing number(s) associated with a dialed 800, 888, or 900 number. The SCP uses TCAP to return a response containing the routing number(s) back to the SSP.

Line Interface Database (LIDB) application

Calling card calls are validated and billed using TCAP query and response messages.

Wireless Application

TCAP provides the ability for automatic roaming service by transferring signaling information to remote nodes. When a mobile subscriber roams into a new mobile switching center (MSC) area, the integrated visitor location register (VLR) requests information such as the service profile from the subscriber's home location register (HLR). In this case, mobile application part (MAP) information is carried within TCAP messages.

TCAP messages are contained within the SCCP portion of a mobile subscriber unit (MSU). An ANSI TCAP message is comprised of a transaction portion and a component portion. In addition, an ITU TCAP (White Book) message contains a dialog portion.

Supported Applications

The addition of the SCCP and TCAP layers being integrated into the SS7 solution allows users to provide enhanced calling features, AIN, and wireless applications. The CSP supports ANSI-41 (formerly IS-41) the mobile phone protocol in U.S. networks, which allows wireless roaming. In international networks, the CSP supports the SS7 mobile

application part of the Global System for Mobile Communications standard (GSM MAP). GSM MAP supports wireless applications such as Short Message Services (SMS).

The Personal Communications Service (PCS) market specifies the need for the capability to map a hybrid of variants in one stack. The architecture is designed to support this type of hybrid configuration of variants. For example, the ITU TCAP over ANSI SCCP is used for database management in the reallocation of calling traffic.

The CSP can be an integral part of an MSC in a PCS Network Architecture, supplying MTP, SCCP and TCAP in order to support an ANSI-41 (formerly IS-41) application. There can be other applications. The MSC, in a mobile switching center, provides services to the mobile subscriber based on information, such as profile information requested by the integrated VLR from the subscriber's HLR.

Two examples of applications for SCCP/TCAP are shown next.

SCCP/TCAP Application Scenarios:

Figure 6-2 Intelligent Network Architecture with SCCP/TCAP

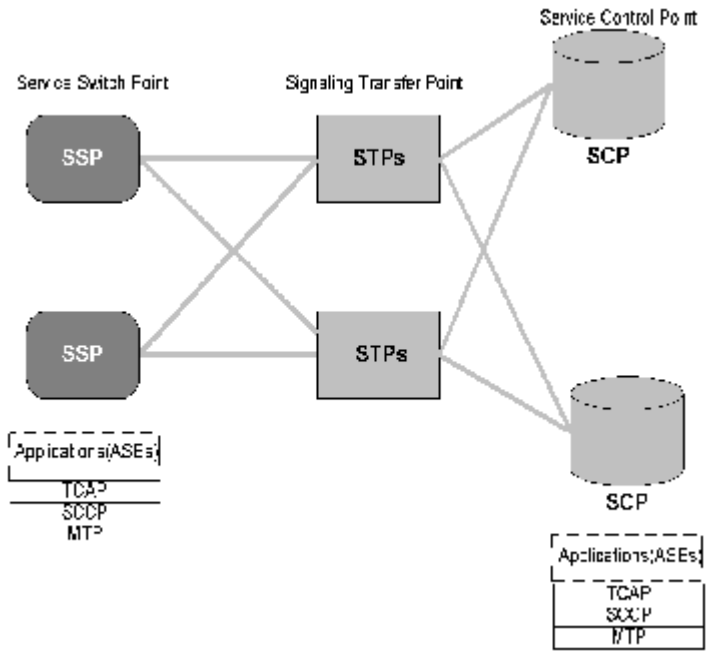
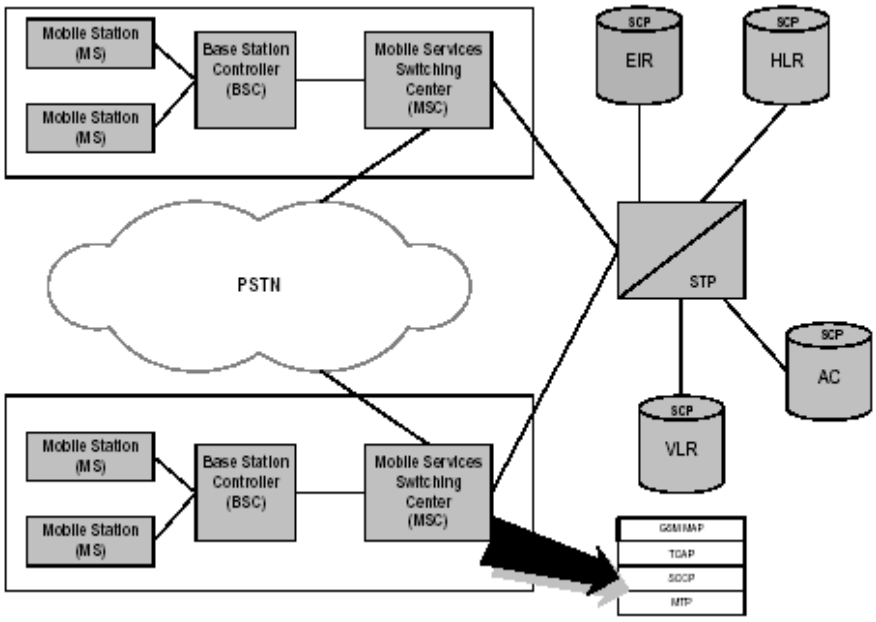


Figure 6-3 Personal Communication Service (PCS) Network



User Interface Requirements

The host has access to the TCAP and SCCP layers through the use of the *PPL Event Request* and *PPL Event Indication* API messages. Both ITU and ANSI variants of SCCP and TCAP primitives are provided.

The ITU primitives are consistent with the ITU White Book Q.771 SCCP and TCAP specifications.

The ANSI primitives are consistent with ANSI T1.1.114-1992 (TCAP) and ANSI T1.112-1996 (SCCP).

Global Title Translation

Overview

As networks grow and become more complex and cross international boundaries, routing between different nodes becomes more complex. One way to simplify this routing requirement is to implement Global Title Translation (GTT), which routes according to a digit analysis rather than Point Code (PC) and Subsystem Number (SSN).

Supports Advanced Features

Dialogic's Global Title Translation (GTT) feature integrated on the CSP provides the routing functionality required to offer services with advanced features such as local number portability (LNP), toll-free, calling card, calling name delivery, and roaming support, as well as other advanced network services.

Capacity Information

The GTT feature supports the following:

- 128 GT groups
- 100,000 entries
- maximum of 24 digits

How Invoked

The GTT functionality enables the Signaling Connection Control Part (SCCP) to translate and route the SCCP message in cases where the Called Party Address contains a Global Title and the routing indicator is set to "Route on GT". GTT functionality also enables the SCCP to have relay functionality. The GTT function is invoked within the SCCP Routing Control (SCRC) under the routing procedures. The Global Title Addressing and GTT feature will allow diverse groups of SCCP addressable entities associated with different applications to establish their own addressing schemes.

Redundancy

The GTT feature supports the same level of redundancy as all other levels of SS7 functionality implemented on the CSP.

Hardware Requirements

This GTT implementation on the CSP requires the SS7 Series 3 card.

Limitations

GTT implementation on the CSP is not configurable on the SS7 PQ card.

Configuring Global Title Translation

Refer to the following sections for information to configure Global Title Translation.

- *Procedures for configuring options (6-17)*
- *Global Title Translation (GTT) Configuration Samples (6-22)*

API Messages

The following API messages support Global Title Translation. They are documented in the *API Reference.*

- *SS7 SCCP/TCAP Configure* (0x0077)
- *SS7 SCCP/TCAP Query* (0x0078)

TLVs

The following Tag Length Values (TLVs) support Global Title Translation. They are documented in the *API Reference.*

- *0x09CA Add Global Title Group*
- *0x09CB Delete Global Title Group*
- *0x09CC Add Global Title Entry*
- *0x09CD Delete Global Title Entry*
- *0x09CE Build Index Table*
- *0x09CF Global Title Group Query*
- *0x09D0 Global Title Entry Query*
- *0x09D1 Global Title Group Query Response*
- *0x09D2 Global Title Entry Query Response*

Message Flow of TCAP Components

Overview

This section contains the SS7 SCCP and TCAP PPL components and their IDs. Also, you will find diagrams showing the message flow for SCCP and TCAP components.

SCCP PPL Components

The following list contains the SS7 SCCP PPL Components and their IDs:

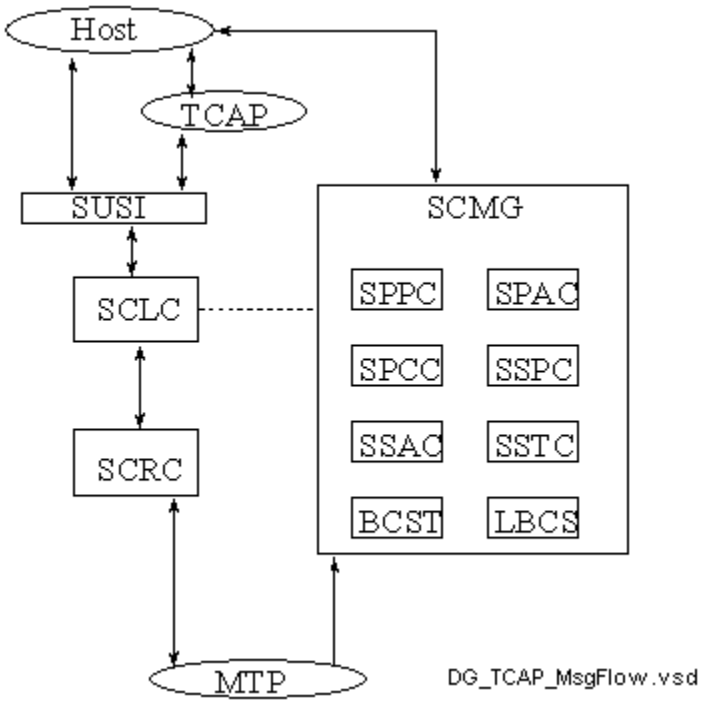
Component ID	Component Name
0x65	SCLC - SCCP Connectionless Control
0x66	SCRC - SCCP Routing Control
0x67	SUSI - SCCP User Interface
0x68	SPPC - Signaling Point Prohibited Control
0x69	SPAC - Signaling Point Allowed Control
0x6A	SPCC - Signaling Point Congestion Control
0x6B	SSPC - Subsystem Prohibited Control
0x6C	SSAC - Subsystem Allowed Control
0x6D	SSTC - Subsystem Status Test Control
0x6E	BCST - Broadcast
0x6F	LBCS - Local Broadcast

TCAP PPL Components

Component ID	Component Name
0x70	TUSI - TCAP User Interface
0x71	CCO - Component Portion Control
0x72	ISM - Component Portion
0x73	TCO - Transaction Coordinator
0x74	TSM - Transaction State Machine
0x75	DHA - dialog Handling

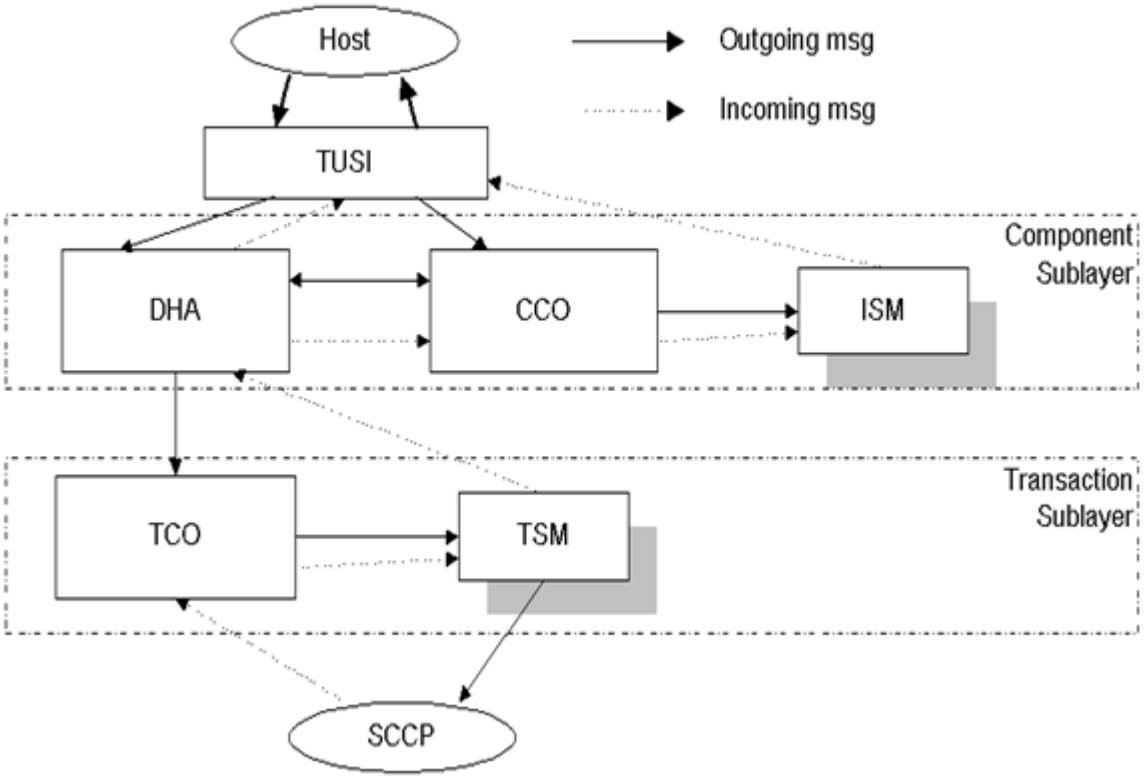
Message flow for SCCP and TCAP components

Figure 6-4 SCCP Components and Message Flow



Note: SCMG = SCCP Management

Figure 6-5 TCAP Components and Message Flow


Host
Outgoing msg
Incoming msg
TUSI
Component Sublayer
DHA
CCO
ISM
Transaction Sublayer
TCO
TSM
SCCP

SCCP/TCAP Call Flows

Overview

This section shows examples of normal and abnormal SCCP/TCAP message flows. These diagrams are intended to show the protocol aspect of the flow correlating with the host messages. It is not a complete list to cover all the cases. More complete SCCP and TCAP call flows can be found in ITU SCCP and TCAP test specifications.

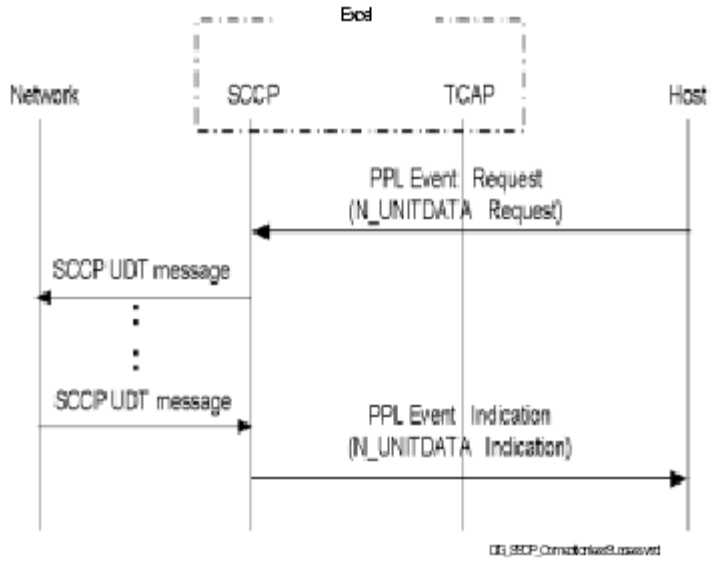
Host Using SCCP Connectionless Service Directly

These call flows show the normal and abnormal cases when the host uses the SCCP service directly (TCAP service is not used). Only the SCCP peer level service is shown here.

Success

SCCP successfully sends the UNIT-DATA message to the network and informs the host about SCCP receiving the Unitdata message from the network.

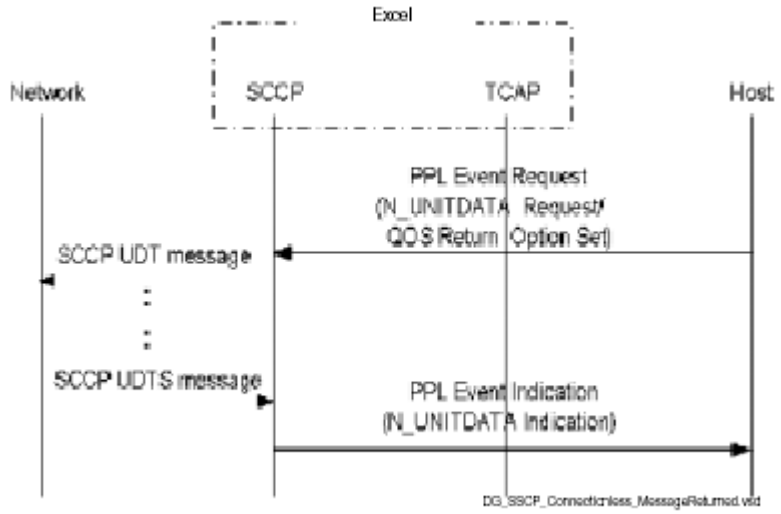
SCCP Connectionless Service Direct - Success



Message Returned

SCCP sends the UNIT-DATA message to the network and sets the return option in the Quality of Service parameter. In this case, the message cannot be delivered to the destination due to failures on the network (such as, a Global Title is translated to an unavailable DPC). The message is returned and the host is informed by an N_NOTICE Indication primitive in a *PPL Event Indication* message.

SCCP Connectionless Service Direct - Message Returned



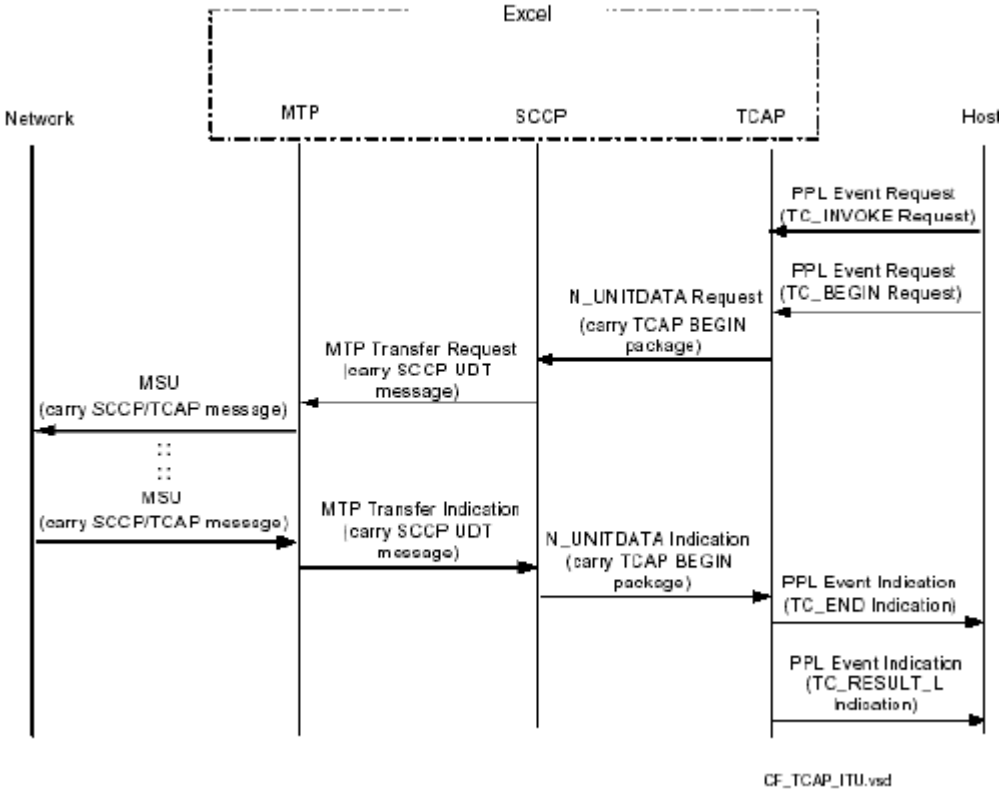
Host Using ITU TCAP Service

The following diagrams show examples of the host using TCAP services. Note that using TCAP service implies the use of SCCP service in the current implementation.

One Transaction/ One Invocation

The services of each SS7 protocol level is used when the host initiates a single transaction with a single remote invocation. In this case, the transaction is successfully completed and the remote operation is successfully invoked.

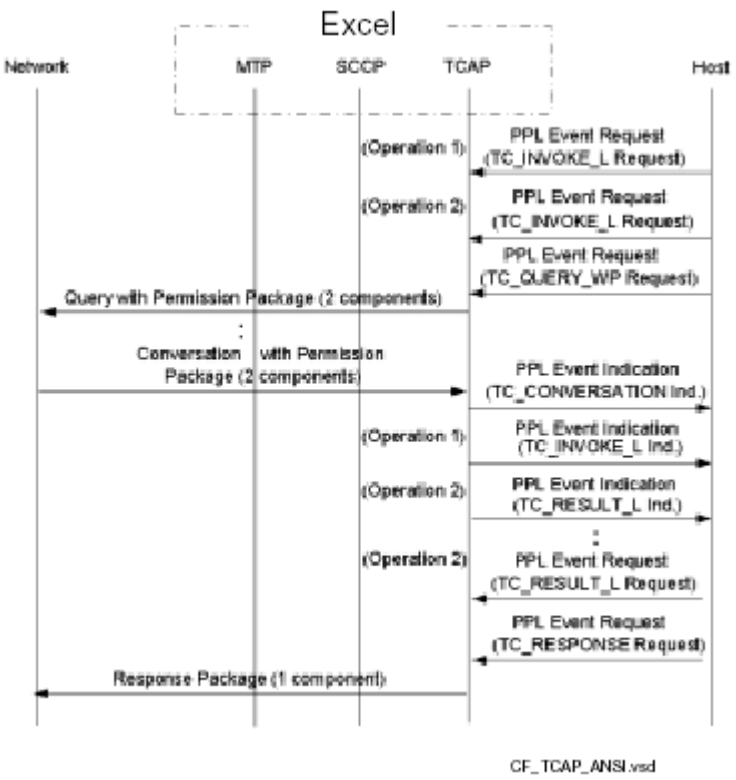
Host Using TCAP Service - 1 Transaction, 1 Invocation (ITU)



One Transaction, Two Invocations

The host uses ANSI SCCP/ TCAP services, while one transaction is created and completed successfully, two remote operations are successfully invoked. Only TCAP peer-level services are shown. The SCCP and MTP services are similar to the previous example.

Host Using TCAP Service - 1 Transaction, 2 Invocations (ANSI)



Configuring SCCP TCAP

Overview

This section describes the SCCP TCAP options that can be configured for a variant of an ANSI or ITU protocol.

Before you begin

Before you can configure other options, you must add a local subsystem and configure how it routes network messages (either to TCAP or to the host). You must use the *SS7 SCCP/TCAP Configure* message to set the byte for Config Type to 0x01 Local SSN.

> **Important!** When using SS7 PQ cards in either a single node or a multi-node CSP system you may configure a total of 3000 physical and virtual CICs, if both ISUP and SCCP/TCAP are to be configured for this system. It does not matter what combination there is of physical or virtual CICs since both require memory space on the card.

Configuring SCCP/TCAP

Use the steps below to configure SCCP/TCAP.

1. Configure the stack using the *SS7 Stack Configure* message.

2. Configure MTP for the stack (link sets, links, routes) using the following messages:
   - *SS7 Signaling Route Configure*
   - *SS7 Signaling Link Set Configure*
   - *SS7 Signaling Link Configure*

3. Configure the local subsystem number using the *SS7 SCCP/TCAP Configure* message.

4. Next you can configure SCCP/TCAP options described in the next section.

END OF STEPS

Procedures for configuring options

You can configure the following SCCP/TCAP options.

Option	Use the SS7 *SCCP/TCAP Configure* message to...
Adjacent Translator	Set the Config Type byte to 0x02 and configure all Adjacent Translators for a given subsystem. By default, all Adjacent Translators are Concerned Point Codes. See *Adjacent Translators Configuration (6-30).*
Other Concerned Point Code	Set the Config Type byte to 0x03. For an SSP or SCP, Point Codes which must be informed about local (CSP) subsystem status changes are referred to as Concerned Point Codes.
SCCP/TCAP Default Parameter Table	Set the Config Type byte to 0x04. This table contains the default parameters used by SCCP and TCAP for all subsystems associated with a stack. These values are used for Mandatory parameters if not included in a primitive. The following parameters are stored by default, according to the ITU white book. • TCAP Dialog_As_ID • TCAP Unidialog_As_ID • TCAP Protocol Version You do not need to configure this table unless you require modifications.
Subsystem (Based on default parameter table)	Configure the default parameter table for a particular sub-system. Set the Config Type byte to 0x05.
Configure Local SSN	Set the Config Type byte to 0x01. You may route messages to TCAP or the host by configuring the SSN Routing Flag.
Network DPC/SSN	Set the Config Type byte to 0x07 to configure the Destination Point Code (DPC) and remote subsystem number.
SCCP/TCAP host configuration	Set the Config Type byte to 0x08 to configure the SCCP/TCAP host that is used to receive messages from the SCCP/TCAP layer.
Configure Replicated SSN	Set the Config Type to 0x09 to configure the Replicated Subsystem Number.
Global Title Translation	Set the Config Type byte to 0x0C to configure Global Title Translation using the TLV Format Configuration Contents. See Global Title Translation (GTT) Configuration Samples.

Querying Use the SS7 *SCCP/TCAP Query* message to query the following:

- Local Subsystem Table
- SCCP Default Parameter Table

- Subsystem Default Parameter Table
- TCAP Active Transactions
- TCAP Active Operations
- SCCP Replicated SSN
- Associated subsystems for a stack
- Number of active dialogs and invocations
- TLV Format Configuration Contents

Deconfiguration

To deconfigure an SS7 stack with SCCP/TCAP, do the following:

1 Put all SSNs out of service using the N-STATE request. This will abort any active TCAP transactions for the SSN if TCAP TUSI Config Byte 3 is set to “Yes” (default value).

2 Deconfigure the SSN using the *SS7 SCCP/TCAP Configure* message.

3 Deconfigure MTP for the stack (links, link sets, route) using the following messages:

- SS7 Signaling Link Configure
- SS7 Signaling Link Set Configure
- SS7 Signaling Route Configure

4 Deconfigure the stack using the *SS7 Stack Configure* message.

END OF STEPS

China National Variant Configuration

Overview

This section describes a configuration sequence to enable MTP/ SCCP/ TCAP support for the China variant. Note that the MTP PPL Config Byte values are listed in the *SS7 PPL Information*, Chapter 6.

> **Important!** National variants may require additional User Part and protocol modifications.

Configuration sequence

The following software configuration sequence (which does not have to be followed in the order presented) should be used to support the SS7 stack for SCCP/TCAP:

- Modify PPL Config Bytes with the *PPL Configure* message:

  MTP3 HMDT component (0x002B)
- Set Config Byte 1 (Variant) to 0x03 (China)

  MTP3 HMRT component (0x002C)
- Set Config Byte 1 (Variant) to 0x03 (China)

  SCCP SCLC component (0x0065)
- Set Config Byte 1 (Variant) to 0x02 (China)

Example

In a configuration with two SS7 stacks (Stack 00 and Stack 01), the following *PPL Configure* messages would be sent to the CSP. The shaded columns (asterisk* for html documents) indicate modifications. Abbreviations used in the table are:

- Seq. No. - Sequence Number
- LNI - Logical Node ID
- PPL Comp. ID - PPL Component ID

Length (MSB, LSB)	Message Type (MSB, LSB)	Reserve Byte	Seq. No.	LNI	AIB	Stack*	PPL Comp. ID*	PPL Entity	# Data Location	Byte	Data*
00 10	00 D7	00	00	FF	00 01 18 01	00	00 2B	01	01	01	03
00 10	00 D7	00	00	FF	0001 18 01	00	00 2C	01	01	01	03

Length (MSB, LSB)	Message Type (MSB, LSB)	Reserve Byte	Seq. No.	LNI	AIB	Stack*	PPL Comp. ID*	PPL Entity	# Data Location	Byte	Data*
00 10	00 D7	00	00	FF	0001 18 01	00	00 65	01	01	01	02
00 10	00 D7	00	00	FF	0001 18 01	01	00 2B	01	01	01	03
00 10	00 D7	00	00	FF	0001 18 01	01	00 2C	01	01	01	03
00 10	00 D7	00	00	FF	0001 18 01	01	00 65	01	01	01	02

Global Title Translation (GTT) Configuration Samples

Purpose

This section contains sample messages for configuring Global Title Translation (GTT). (These samples are easier to read in the PDF format of this document rather than the web-based format.)

Using these sample messages

The description for the TLVs precede each group of samples. The different options that you can select with each Tag Length Value (TLV) are included in the sample messages.

API Messages

Refer to the following two EXS API messages in the *API Reference* for for details on configuring GTT.

- *SS7 SCCP/TCAP Configure (0x0077)*
- *SS7 SCCP/TCAP Query (0x0078)*

Adding Groups

Use the following TLV to add Global Title Groups.

Table 6-1 Add Global Title Group TLV 0x09CA

Byte	Description	
0, 1	Tag	0x09CA Add Global Title Group
2	Length	0x0009
3	Value[0]	0-127 Group ID (GID)
4	Value[1]	Global Title Indicator (GTI) 1-4 for ITU 1-2 for ANSI
5	Value [2]	Translation Type (TT) 0-255 (0 = Default)
6	Value [3]	Numbering Plan (NP) 0-15 (0 = Default)
7	Value [4]	Nature of Address Indicator (NAI) 1-127 (0 = Default)
8	Value [5]	Minimum Digits (1-24) (MIN) The minimum number of digits that the GT entry of this group will contain. Must be greater than 0 and less than or equal to the MaximumDigits field.
9	Value [6]	Maximum Digits (1-24) (MAX) The maximum number of digits that the GT entry of this group will contain. Must be less than 24 and greater than or equal to the Minimum Digits field.

10	Value [7] Group Attribute (ATT) Bit 0 Maximum match 0 = Use minimum match rule for this group 1 = Use maximum match rule (also known as best match rule) for this group. Bit 1 Calling Party Address (CGPA) treatment 0 = Calling Party address will not be changed when the Global Title Global Title translation results in “Route on GT.” 1 = Calling Party address is changed whenGT translation results in “Routeon GT.” The new calling party address is stored in SSN default parameter table. The CGPA bit will not take effect for connection oriented messages. Bit 2 ANSI TT4 (Translation Type 4) 0 = Not TT4 type 1 = TT4 type If this bit is set, no GT entry can be added into this group. Bit 3 Remove GT 0 = Do not remove GT after a successful translation. 1 = Remove GT after a successful translation.
11	Reserved for future use. (REV)

Sample 1

The Group Attribute 08 indicates use the Minimum Match Rule and Remove the GT after Translation.

```
          Add group         AIB        STK     TYP     TAG     LEN     GID GTI TT  NP NAI MIN MAX
    ATT         REV
00 19 00 77 00 00 FF 00 01 08  01 00   0C   01 09 CA  00 09  00  04  01  04 03  01  18  08  00
```

Sample 2

The Group Attribute 00 indicates use the Minimum Match Rule and Do Not Remove GT after Translation.

```
          Add group         AIB     STK   TYP   TAG       LEN  GID GTI TT NP NAI MIN MAX ATT REV
00 19 00 77 00 00 FF 00 01 08 01 00 0C 01 09 CA 00 09 01   04 02 04 03 01 18  00  00
```

Sample 3

The Group Attribute 09 indicates use the Maximum Match rule and Remove GT after Translation.

```
       Add group           AIB      STK  TYP TAG       LEN   GID GTI TT NP NAI MIN MAX ATT REV
00 19 00 77 00 00 FF 00 01 08 01 00 0C  01 09 CA 00 09 02  04 03 04 03 01 18  09  00
```

Sample 4

The Group Attribute 01 indicates use the Maximum Match Rule and Do Not Remove GT after Translation.

```
      Add group           AIB     STK   TYP   TAG      LEN   GID GTI TT NP NAI MIN MAX ATT REV
00 19 00 77 00 00 FF 00 01 08 01 00 0C  01 09 CA 00 09 03  04 04 04 03 01 18  01  00
```

Important! The index table is rebuilt automatically for the GT Group operation (Add/Delete).

Deleting Groups

Use the following TLV to delete groups identified by the Global Title Indicator, Translation Tile, Numbering Plan, and Nature of Address Indicator.

Table 6-2 Delete Global Title Group TLV 0x09CB

Byte	Description	
0, 1	Tag	0x09CB Delete Global Title Group
2	Length	0x0006
3	Value[0]	Group ID (GID) 1-127
4	Value [1]	Global Title Indicator (GTI) 1-15
5	Value [2]	Translation Type (TT) 0-255 (0- Default)
6	Value [3]	Numbering Plan (NP) 0-15 (0 - Default)
7	Value [4]	Nature of Address Indicator (NAI) 1-127 (0 - Default)
8	Reserved for future use. (REV)	

The sample below deletes four groups: 00 to 03.

```
 Delete Group      AIB    STK   TYP     TAG    LEN   GID  GTI   TT  NP  NAI REV
00 16 00 77 00 00 FF     00 01 08   01 00       0C          01 09 CB    00 06     00      04
    01  04    03    00

00 16 00 77 00 00 FF     00 01 08    01 00      0C          01 09 CB   00 06      01      04
 02  04    03    00

00 16 00 77 00 00 FF     00 01 08    01 00      0C          01 09 CB   00 06      02      04
 03  04    03    00

00 16 00 77 00 00 FF     00 01 08    01 00      0C          01 09 CB   00 06      03      04
 04  04    03    00
```

Important! The index table is rebuilt automatically.

Adding GT Entries

Use the following TLV to add Global Title Entries.

Table 6-3 Add Global Title Entry TLV 0x09CC

Byte	Description	
0, 1	Tag	0x09CC Add Global Title Entry
2, 3	Length	Variable
4	Value[0]	Group ID (GID) 1-127
5	Value [1]	Entry Attribute Bit 0 - Wild Card bit 0 - Non Wild Card GT 1 - Wild Card GT Bits 1-7 Not Used.
6	Reserved for future use.	
7	Value [2]	Global Title Address Information Length (LEN) 1-24 digits
:	Value [3-n]	Global Title Address Information (GTAI) See *API Reference* for format.
:	Value [n]	Translation Result Option 00 - Single translation result 01 - Double translation results, working in Active Standby Mode 02 - Double translation results, working in Load Sharing Mode
:	Value [n]	Translation Result 1 See in the *API Reference* for format.

Sample 1

The following values apply to the sample below:

- Entry Attribute: Non wild card entry
- Translation result format: RI+PC+SSN (RI=Routing Indicator, PC=Point Code, SSN=Subsystem Number)
- RI= Route on DPC/SSN

```
   Add GT Entry    AIB   STK    TYP  TAG  LEN GID ATT REV LEN  GTAI
00 28 00 77 00 00 FF 00 01 08 01 00        0C       01 09 CC  00 18    02    00     00     16
   80 01 23 41 23 45 67 89 12 34 01\

GTAI (Cont)
00 07 01 01 00 00 02 22 05
```

Sample 2

The following values apply to the sample below:

- Entry Attribute: Wild card entry
- Translation result format: RI+PC+SSN
- RI= Route on DPC/SSN

```
   Add GT Entry                  STK  TYP      TAG   LEN  GID ATT REV LEN        GTAI
00 1e 00 77 00 00 FF 00 01 08 01 00   0C  01 09CC00 0E   02  01  00  02      80 00 07 01 01 00
   00 02 22 05
```

Sample 3

The following values apply to the sample below:

- Entry Attribute: Non wild card entry
- Translation result format: RI+PC+GT
- RI= Route on GT

```
   Add GT Entry        AIB      STK  TYP      TAG   LEN  GID ATT REV LEN  GTAI
00 35 00 77 00 00 FF 00 01 08 01 00   0C 01  09 CC 00 25  02   00   00       16     80 01 23 41
   23 45 67 89 12 34 07 00 14 02\

GTAI (Cont)
 00 00 00 02 22 01 0c 03 80 01 23 41 23 45 67 89 12 35 01
```

Sample 4

The following values apply to the sample below:

- Entry Attribute: Non wild card entry

- Translation result format: RI+PC
- RI= Route on GT

```
   Add GT Entry           AIB       STK  TYP    TAG   LEN   GID ATT REV LEN  GTAI
00 23 00 77 00 00 FF 00 01 08 01 00   0C 01 09 CC   00 13   02    00     00     0e     80 01
   23 00 00 00 00 00 06 00 00 00 00 02 22
```

Sample 5

The following values apply to the sample below:

- Entry Attribute: Non wild card entry
- Translation result format: RI+PC+SSN+GT
- RI= Route on DPC/SSN

```
            Add GT Entry       AIB     STK   TYP    TAG        LEN   GID ATT REV LEN   GTAI
00 2e 00 77 00 00 FF 00 01 08 01 00    0C      01 09 CC   00 1E  02     00     00    16     8
   01 23 41 23 45 67 89 12 34 09 00 0d 03\

GTAI (Cont.)
   01 00 00 02 22 03 01 04 02 12 30 02
```

Deleting GT Entries

Use the following TLV to delete the GT entries indicated by the Group ID, GT Attribute, Global Title Address Information (GTAI) Length, and GTAI.

Table 6-4 Delete Global Title Entry TLV 0x09CD

Byte	Description	
0, 1	Tag	0x09CD Delete Global Title Entry
2, 3	Length	Variable
4	Value[0]	Group ID 1-127 Already configured.

5	Value [1] Entry Attribute Bit 0 - Wild Card bit 0 = Non Wild Card GT 1 = Wild Card GT Bits 1-7 Not Used.
6	Reserved for future use.
7	Value [2] Global Title Address Information Length (LEN) 1-16 digits
:	Value [n] Global Title Address Information (GTAI)

The sample below deletes five GT entries in GT Group 2.

```
          Delete GT Entries      AIB      STK TYP    TAG          LEN     GID ATT REV LEN    GTAI
00 1f 00 77 00 00 FF 00 01 08 01   00       0C     01 09 CD  00 0f     02      00    00     16
    80 01 23 41 23 45 67 89 12 34 07

00 15 00 77 00 00 FF 00 01 08 01  00      0C      01 09 CD   00 05    02      01   00       02
      80

00 1f 00 77 00 00 FF 00 01  08 01  00      0C      01 09 CD   00 0f     02     00   00       16
   80 01 23 41 23 45 67 89 12 34 07

00 1b 00 77 00 00 FF 00 01 08 01  00      0C      01 09 CD    00 0b   02      00   00       0e
   80 01 23 00 00 00 00

00 1f 00 77 00 00 FF 00 01 08 01  00      0C     01 09  CD     00 0f    02     00    00       16
   80 01 23 41 23 45 67 89 12 34 09
```

Building the Index Table

Use this TLV to build the index table which brings the new GT entries into service and takes the deleted GT entries out of service.

Table 6-5 Build Index Table TLV 0x09CE

Byte	Description	
0, 1	Tag	0x09CE Build Index Table
2, 3	Length	0x0001
4	Value[0]	Bit 1 - 1=Build Index

```
LEN     VALUE
CE  00 01     01
```

```
     Build Index Table          AIB          STK  TYP    TAG
00 11 00 77 00 00 FF     00 01 08 01  00       0C      01 09
```

Adjacent Translators Configuration

Overview

In SCCP, a Global Title Translator is the node that performs the global title translation (GTT). The result of the GTT could be the DPC/SSN or another global title, in which case another node is used to perform further GTT. Therefore, there may be more than a translator involved before an SCCP message reaches its destination.

Adjacent Translator

The Adjacent Translator is the first node that performs the GTT from the point of view of the SSP (adjacent is a logical term). For example, Node A is adjacent to Node B if any SS7 path exists between them without a GTT in the path.

Alternatively, the physically adjacent STP could be the Adjacent Translator and be responsible for routing to the next GTT. Dialogic's SCCP implementation places no limitation on the network configuration.

If an Adjacent Translator needs to be defined for a subsystem, the route to the Point Code must be defined first. If it is not, a NACK of DPC Not Configured (0x550E) is returned. To delete a route, you must first delete the Adjacent Translator, otherwise, a NACK of SCCP/TCAP Related Configuration Parameter Error (0x55).

If you delete an SSN, all of its associated configuration, including Adjacent Translators, is automatically deleted.

Routing with a Global Title

For any local SSN, you can define Adjacent Translators and Network destination point codes and subsystem numbers. If the adjacent translator is defined, you may route the message to the translator node for global title translation

Example Configurations and TCAP API Messaging

Overview

Four sample configuration files are included with the release software showing the steps required for basic and SCCP/TCAP-specific configuration. These files are shown further on in this section.

Descriptions of Sample *.cfg Files

The sample configuration is for one node with two SS7 stacks in loop-back mode. Each stack simulates an SS7 node. The sample configuration files are summarized below:

- "itumtp.cfg" includes the configuration for the two SS7 stacks (including the SCCP/TCAP module) and MTP configuration. The following configuration is performed:
  - SS7 spans
  - signaling stacks, link sets, links, and routes
  - bring spans, channels, and links in service
- "sccp.cfg" includes SCCP-related configuration information. There are two subsystems configured for each stack. The following configuration is performed:
  - SCCP local SSN
  - Adjacent translators
  - Other Concerned Point Codes (optional)
  - Bring SSNs in service with N_STATE Request in *PPL Event Request* message.
  - Network DPC/SSN (for direct routing to a network SSN)
- "unitdata.cfg" is an example of using SCCP N-UNITDATA service.
- "begin.cfg" is an example of using TCAP service by invoking the ITU TC_INVOKE and TC_BEGIN primitives. ITUMTP.CFG -- Sample Configuration File

itumtp.cfg

' de-assign all spans and assign logical spans

```
00 0d 00 a8 00 00 ff  00 01 11 04 ff ff ff ff
00 0d 00 a8 00 00 ff 00 01 11 04 00 00 04 00
00 0d 00 a8 00 00 ff 00 01 11 04 00 01 04 01
00 0d 00 a8 00 00 ff 00 01 11 04 00 02 04 02
00 0d 00 a8 00 00 ff 00 01 11 04 00 03 04 03
00 0d 00 a8 00 00 ff 00 01 11 04 00 04 04 04
00 0d 00 a8 00 00 ff 00 01 11 04 00 05 04 05
00 0d 00 a8 00 00 ff 00 01 11 04 00 06 04 06
00 0d 00 a8 00 00 ff 00 01 11 04 00 07 04 07
```

' Configure SS7 Spans (T1)

```
00 0d 00 a9 00 00 01 00 01 0c 02 00 00 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 01 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 02 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 03 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 04 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 05 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 06 52 06
00 0d 00 a9 00 00 01 00 01 0c 02 00 07 52 06
```

' Configure First SS7 Stack (A)

```
' Set Single Redundancy on Board 2

00 0d 00 5b 00 00 ff 00 02 01 01 02 01 01 ff
```

'Set Signaling Stack Configure A

```
' Set our OPC to 00000111 and set the 5 modules to ITU (MTP,ISUP,L3P,SCCP,TCAP)

00 1a 00 5c 00 00 ff 00 01 21 02 02  00  00 00 01 11 05 01 01 02 01 03 01 06 01 07 01
```

' Set the Signaling Link Set Config. A

```
' Define a Link Set going to 00000222 ID 0
00 0f 00 5d 00 00 ff 00 01 1e 02 00  00 00 00 02 22
```

' Set the Signaling Link Config. A

```
' Define a link ID 0 for Set-0 SLC-0 Using Span/Channel (0,0) 64k
' Define a link ID 1 for Set-0 SLC-1 Using Span/Channel (2,0) 64k
00 14 00 5e 00 00 ff 00 02 1f 03 00  00 01 0d 03 00 00 00 00 00 00
00 14 00 5e 00 00 ff 00 02 1f 03 00  00 02 0d 03 00 01 00 01 00 00
```

Set the Signaling Route Configure for A '

```
                          ' Define a Route to 00 00 02 22 using only Link Set 0
'                                                  p
'                                               l  r
'                                               i  i
'                       s                       n  o
'                       t                       k  r
'                       a                       s  i
'                       c                       e  t
'                       k  [dst] [rte] [   dpc   ] t  y
00 14 00 5f 00 00 ff 00 01 20 05 00 00 00 00 00 00 00 02 22 00 00
```

' Configure Second SS7 Stack (B)

```
'Set Signaling Stack Configure B
' Set our OPC to 00000222 and set the 5 modules to ITU(MTP,ISUP,L3P,SCCP,TCAP)
'        brd
00 1a 00 5c 00 00 ff 00 01 21 02 02 01 00 00 02 22 05 01 01 02 01 03 01 06 01 07 01

                          ' Set the Signaling Link Set Config. B
                          ' Define a Link Set going to 00000111 ID 1
                          00 0f 00 5d 00 00 ff 00 01 1e 02 01  01 00  00 01 11

' Set the Signaling Link Config. B
' Define a link ID 1 for Set-1 SLC-0 Using logical Span (63,0) 64k
00 14 00 5e 00 00 ff 00 02 1f 03 01 01  02 0d 03 00 01 00 00 00 00
00 14 00 5e 00 00 ff 00 02 1f 03 01 01  03 0d 03 00 03 00 01 00 00

                          ' Set the Signaling Route Configure
                          ' Define a Route to 00 00 01 11 using only Link Set 1


                '                                              p
                '                                           l  r
                '                                           i  i
                '                       s                   n  o
                '                       t                   k  r
                '                       a                   s  i
                '                       c                   e  t
                '                       k  [dst] [rte] [   dpc   ] t  y
               00 14 00 5f 00 00 ff 00 01 20 05 01 00 01 00 01 00 00 01 11 01 00

'
```

' Service State Configure

```
‘Place All Spans and Channels In service.
' 00 0d 0a 00 00 ff <ENTITY> <ACTION> <ID0> <ID1> <ID2> <ID3> <FORCE>

                          00 0d 00 0a 00 00 ff 00 01 0c 02 00 00 F0 00
                          00 0d 00 0a 00 00 ff 00 01 0c 02 00 01 F0 00
                          00 0d 00 0a 00 00 ff 00 01 0c 02 00 02 F0 00
                          00 0d 00 0a 00 00 ff 00 01 0c 02 00 03 F0 00
```

'Bring links Into Service

```
                          ' Bring the Links in Service
                          00 0d 00 0a 00 00 ff 00 01 09 02 00 00 f0 00
                          00 0d 00 0a 00 00 ff 00 01 09 02 00 01 f0 00
                          00 0d 00 0a 00 00 ff 00 01 09 02 01 02 f0 00
```

```
                    00 0d 00 0a 00 00 ff 00 01 09 02 01 03 f0 00
```

sccp.cfg

```
'configure SCCP local SSN


'config local subsystem 2 and 5 for stack 0, subsystem 3 and 6 for stack 1.
'Subsystem 2 and 3 use sccp service only
'Subsystem 5 and 6 use tcap/sccp service.
                          '  AIB Stack CfgType SSN  Option Use_tcap Reserve
00 0f 00 77 00 00 ff 00 01 08 01 00    01      02   01     00       00
00 0f 00 77 00 00 ff 00 01 08 01 00    01      05   01     01       00
00 0f 00 77 00 00 ff 00 01 08 01 01    01      03   01     00       00
00 0f 00 77 00 00 ff 00 01 08 01 01    01      06   01     01       00
'config adjacent translator
'Config spc 222 as the adjacent translator for subsystem 2 and 5
'Config spc 111 as the adjacent translator for subsystem 3 and 6
'                         AIB     Stk   CfgType SSN  Opt    DPC
00 0f 00 77 00 00 ff 00 01 08 01 00    02      02   01     00 00 02 22
00 0f 00 77 00 00 ff 00 01 08 01 00    02      05   01     00 00 02 22
00 0f 00 77 00 00 ff 00 01 08 01 01    02      03   01     00 00 01 11
00 0f 00 77 00 00 ff 00 01 08 01 01    02      06   01     00 00 01 11

'config other concerned pc, (can be used as user's option, it is not used in
' this example)
' it can be informed the subsystem 2 status if changed.
'                         AIB     Stk   CfgType SSN  Opt    DPC
00 11 00 77 00 00 ff 00 01 08 01 00    03      02   01 00 00 02 22
00 11 00 77 00 00 ff 00 01 08 01 00    03      05   01 00 00 02 22
00 11 00 77 00 00 ff 00 01 08 01 01    03      03   01 00 00 01 11
00 11 00 77 00 00 ff 00 01 08 01 01    03      06   01 00 00 01 11

'put all the ssn in service, using N_STATE req in ppl event req
' 01
```

```
                                  ------------------------------------------------
'                                    AIB           comp  ppl-ev icb-cnt
00 1e 00 44  00 00  ff  00 01 2a 03 00 02 00 00 67 00 03 01 02 20 0a  00 6e 00 01 02
   00 70 00 01 01
00 1e 00 44  00 00  ff  00 01 2a 03 00 05 00 00 67 00 03 01 02 20 0a  00 6e 00 01 05
   00 70 00 01 01
00 1e 00 44  00 00  ff  00 01 2a 03 01 03 00 00 67 00 03 01 02 20 0a  00 6e 00 01 03
   00 70 00 01 01

'config network dpc-ssn. Use this config if a certain subsystem needs to talk to
' a network dpc/ssn directly (route on dpc/ssn). You do not need to config this
' if route on global title. A network dpc/ssn could be a remote or local
' subsystem.
'                          AIB     Stk   CfgType SSN  Opt    DPC     SSN
00 12 00 77 00 00 ff 00 01 08 01 00    07      02   01 00 00 02 22 03
00 12 00 77 00 00 ff 00 01 08 01 00    07      02   01 00 00 02 22 06
00 12 00 77 00 00 ff 00 01 08 01 00    07      05   01 00 00 02 22 03
00 12 00 77 00 00 ff 00 01 08 01 00    07      05   01 00 00 02 22 06
00 12 00 77 00 00 ff 00 01 08 01 01    07      03   01 00 00 01 11 02
00 12 00 77 00 00 ff 00 01 08 01 01    07      03   01 00 00 01 11 02
00 12 00 77 00 00 ff 00 01 08 01 01    07      06   01 00 00 01 11 05
00 12 00 77 00 00 ff 00 01 08 01 01    07      06   01 00 00 01 11 05
```

unitdata.cfg

```
'N_UNITDATA request, with return option set, sequence not guaranteed.
'stack 0 subsystem 2 send msg to stack 1 subsystem 3, route on dpc/ssn
'                                comp  event ICB       CGPA
   CDPA                                          QOS                 user data
00 3e 00 44 00 00 ff 00 01 2a 03 00 02 00 00 67 00 01 01 02 20 2a 00 67 00 09 01 00
   02 00 00 01 11 00 00 00 69 00 09 01 00 03 00 00 02 22 00 00 00 6a 00 03 01 00 00
   00 6b 00 05 11 22 33 44 55
'N_UNITDATA request, with return option set, sequence not guaranteed.
'stack 0 subsystem 02 send msg to adjacent translator 0x333, route on global title
'this is supported as user's option, it is not used in this example configuration
'                                comp  event ICB       CGPA
   CDPA                                                QOS                 user data
00 41 00 44 00 00 ff 00 01 2a 03 00 02 00 00 67 00 01 01 02 20 2d 00 67 00 09 01 00
   02 00 00 01 11 00 00 00 69 00 0c 00 00 00 00 00 03 33 02 03 12 34 56 00 6a 00 03
   01 00 00 00 6b 00 05 11 22 33 44 55
```

begin.cfg

```
' stack 0 subsystem 5 send TC_INVOKE, dialog id 00 00 00 01
' class 1, invoke id 01, local operation code 1c2b, timeout 120 s, no parameter
'                           AIB            comp  event ICB dialog_id
00 2e 00 44 00 00 ff 00 01 2a 03 00 05 00 00 70 00 15 01 02 21 1a 00 00 00 01
' class invoke_id      operation        timeout
00 28 00 01 01 00 1f 00 01 01 00 2c 00 02 1c 2b 00 29 00 02 2e e0

                           ' TC_BEGIN
                           ' stack 0 subsystem 5 send TC_BEGIN, dialog id 00 00 00
                              01,
                           ' send to dpc 222, subsystem 6, route on dpc/ssn
'               AIB          comp  event ICB sccp_icb_head  CGPA
   CDPA                                    QOS                TCAP_ICB_HEAD   'dialog_id
00 3c 00 44 00 00 ff 00 01 2a 03 00 05 00 00 70 00 01 02  02 20 21        00 67 00 09
   01 00 05 00 00 01 11 00 00 00 69 00 09 01 00 06 00 00 02 22 00 00 00 6a 00 03 00
   00 00 02 21 04         00 00 00 01
```

'directhost.cfg

```
                              'configure local host for ssn 05 stack 00
                              'delete matrix host firstly
'                               AIB          config type ssn opt     matrix host
00 0f 00 77 00   00   ff  00 01 08 01 00  08          05  02      ff     00

                              'then add local host
'                               AIB          config type ssn opt     matrix host
00 0f 00 77 00   00   ff  00 01 08 01 00  08          05  01      fe     00
```

'matrixhost.cfg

```
                                'configure matrix host for ssn 05 stack 00
                              'delete local host firstly
'                               AIB          config type ssn opt     matrix host
00 0f 00 77 00   00   ff  00 01 08 01 00  08          05  02      fe     00

                              'then add matrix host
'                               AIB          config type ssn opt     matrix host
00 0f 00 77 00   00   ff  00 01 08 01 00  08          05  01      ff     00
                             'configure matrix host for ssn 05 stack 00
```

TCAP primitive #1

This example along with the next one, TCAP Primitive #2, show how to map the TCAP Primitive Interface to the TCAP Primitive Set Interface.

-- TC_INVOKE request

-- TC_INVOKE request

```
' len  msgid  seq  node     AIB                TUSI  Invoke   #ICB
00 47  00 44 00 00  01  00 01 2a 03 00 05 00  00 70   00 15    01
'TCAP_Parameter_ICB len    Dialog_id
    02 21           33     00 00 00 03
'TCAP parameter TLVs follow
00 1f 00 01 03
00 2d 00 01 2d
00 2e 00 16 30 14 80 07 17 33 12 34 56 78 90 81 01 00 82 06 13 12 34 56 78 90
00 28 00 01 01
00 29 00 02 0f ff
```

TCAP primitive #2 TC_BEGIN request

```
' len  msgid  seq  node     AIB                TUSI   Begin #ICE
 00 48 00 44 00 01  01  00 01 2a 03 00 05 00  00 70   00 01  02
'TCAP_Parameter_ICB len    Dialog_id
    02 21           14     00 00 00 03
'TCAP parameter TLVs follow
00 38 00 07 04 00 00 01 00 20 02
00 45 00 01 01
'SCCP_Parameter_ICB len (SCCP addresses etc)
 02 20    1d
'SCCP parameter TLVs follow
00 67 00 07 01 01 05 00 00 01 11
00 69 00 07 01 01 06 00 00 02 22
00 6a 00 03 00 00 00
```

```
' len  msgid  seq  node     AIB                TUSI   Begin #ICB
 00 48 00 44 00 01  01  00 01 2a 03 00 05 00  00 70   00 01  02
'TCAP_Parameter_ICB len    Dialog_id
    02 21           14     00 00 00 03
'TCAP parameter TLVs follow
00 38 00 07 04 00 00 01 00 20 02
00 45 00 01 01
'SCCP_Parameter_ICB len (SCCP addresses etc)
 02 20    1d
'SCCP parameter TLVs follow
00 67 00 07 01 01 05 00 00 01 11
00 69 00 07 01 01 06 00 00 02 22
00 6a 00 03 00 00 00
```

TCAP Primitive Set It may be helpful to compare this example to the previous examples, TCAP Primitive #1 and TCAP Primitive #2.

TC_BEGIN primitive set request (with TC_Invoke component)

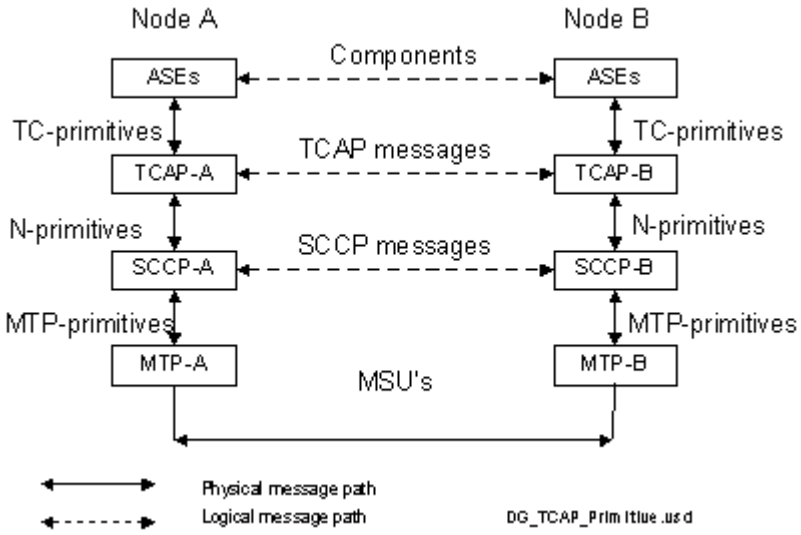
```
' len  msgid  seq  node     AIB                TUSI  Event_id #ICB
00 84  00 44 00 00  01  00 01 2a 03 00 05 00  00 70   00 32    03
'TCAP_Primitive_ICB_#1 len    Dialog_id   Primitive_id(Invoke)
    02 65              35     00 00 00 03     00 15
'TCAP parameter TLVs follow
00 1f 00 01 03
00 2d 00 01 2d
00 2e 00 16 30 14 80 07 17 33 12 34 56 78 90 81 01 00 82 06 13 12 34 56 78 90
00 28 00 01 01
00 29 00 02 0f ff
'TCAP_Primitive_ICB_#2 len    Dialog_id   Primitve_id(Begin)
 02 02 65              16     00 00 00 03     00 01
'TCAP parameter TLVs follow
00 38 00 07 04 00 00 01 00 20 02
00 45 00 01 01
'SCCP_Parameter_ICB len (SCCP addresses etc)
 02 20              1d
'SCCP parameter TLVs follow
00 67 00 07 01 01 05 00 00 01 11
00 69 00 07 01 01 06 00 00 02 22
00 6a 00 03 00 00 00
```

SCCP TCAP Primitives

Overview

The communication that takes place between the MTP, SCCP, and TCAP layers of the SS7 stack is achieved with the use of primitives. Interfaces between the functional elements of SS7 are specified using interface primitives. Primitive interface definition does not assume any specific implementation of a service The host uses SCCP/TCAP services by invoking the SCCP N- or TCAP TC- primitives with the *PPL Event Request* message. A response status of Positive ACK (0x10) indicates that the primitive is validated and processed. A NACK of 0x54 indicates “Error Detected in Primitive”. The LSB of the Status byte gives more detail on the error. TC-USER can re-send the corrected primitive. See Response Status Values in the *API Reference* for details on LSB status.

Figure 6-6 Peer-to-Peer Messages



TCAP Primitive Types

There are two types of TCAP primitives:

- Dialog
- Component

Dialog Primitives

Dialog primitives request or indicate facilities of transaction sub-layer, in relation with message translation or dialog handling. dialog primitives may contain a TCAP ICB or a TCAP ICB and an SCCP ICB.

Component Primitive

Component primitives are used in the handling of operations and replies. They do not require facilities form the underlying sub-layer. Component primitives contain a TCAP ICB.

Dialogs

Simultaneous Transactions

The maximum number of simultaneous transactions is 32,000 per stack. If number is exceeded, any TC-USER request which may result in a new transaction to be created is returned with a NACK of Resource Limitation (0x5409).

You can use the *SS7 SCCP/TCAP Query* message (0x0078) to query the number of active dialog/transaction/operations.

The following may result in a larger number of active dialogs at one time:

- Average life time of a dialog is long.
- Messages that never get a response back (hanging dialog) - these dialogs use the TCAP dialog resources and do not release them.

The maximum number of simultaneous invocations is 96,000. There is no limit on the number of invocations per transaction. If you include multiple components within one TCAP message, you should ensure that it fits in one MSU, otherwise; you may receive a NACK for the message. (If you use the SCCP Segmentation feature of SCCP/TCAP, this limitation does not apply. The maximum length of a TCAP message can be 3092 (ANSI) or 2048 (ITU).)

Important! In this context, incoming dialog means the new dialog is initialized, caused by receiving a network TCAP message (BEGIN[ITU], or QUERY[ANSI]). Outgoing dialog means the new dialog is initialized, caused by the host sending a TCAP primitive (TC_BEGIN[ITU], or TC_QUERY_ with/ without_PERMISSION[ANSI]).

Beginning a dialog

A TC_USER begins a new dialog by issuing a TC-BEGIN (ITU) or TC-QUERY-WITH-PERMISSION/TC-QUERY-WITHOUT-PERMISSION (ANSI) request primitive to:

- indicate to the Component sub-layer that a new dialog starts, identified by the dialog ID parameter.

- request transmission of any component(s) previously passed to the Component sub-layer by means of component handling primitives of the Request type with the same dialog ID.

Continuing a dialog

A TC-USER indicates that it wants to continue a dialog by issuing a TC-CONTINUE (ITU) or TC-CONVERSATION-WITH-PERMISSION/TC-QUERY-WITHOUT-PERMISSION (ANSI) request primitive.

Ending a dialog

There are three methods for ending a dialog:

- Pre-arranged End

The end of the dialog has been decided by prior arrangement of the TC-USERs. The effect of the TC-END (ITU) or TC-RESPONSE (ANSI) request primitive is local only; no TC-END or TC-RESPONSE indication is generated.

- Basic End

The ending causes transmission of any pending components at the side which initiates it.

- Abort by TC-USER

The TC-U-ABORT request and indication primitives are used to indicate abort by the TC-USER.

Dialog ID Usage

The incoming Dialog ID space is a set including 32,000 continuous dialog IDs (per stack). This is because incoming dialog IDs are allocated by TCAP and the maximum TCAP support is 32,000 simultaneous transactions per stack. The beginning of the incoming dialog ID is defined by the TCAP PPL component TCO Config Byte 1-4 (see *TCAP TCO (0x0073*). This can be reconfigured, if necessary. The outgoing dialog ID, however, is specified by the TC user. Even though TCAP software does support any 4-byte dialog ID, as long as it does not collide with the incoming ID set, we recommend the use of the range from 0 - 32767 for performance reasons. The dialog ID, which

must be four bytes, is also used in the SS7 TCAP Parameters (0x21) ICB which is used with the *PPL Event Indication* and *PPL Event Request* API messages.

Exception Reporting and Message Return

TC-USERs are notified of non-delivery of components by the TC-NOTICE indication primitive. A TC-NOTICE indication primitive is only passed to the TC-USER if the requested service cannot be provided and the TC-USER requested the Return Option in the Quality of Service parameter of the dialog handling request primitive.

Invoke a Remote Operation

The TC-INVOKE (ITU), TC-INVOKE-L (ANSI) and TC-INVOKE-NL (ANSI) primitives are used to invoke a remote operation.

- TC-INVOKE-LAST (ANSI)

  Indicates the only or last segment of the invocation.

- TC-INVOKE-NOT LAST (ANSI)

  Indicates a segment of an invocation (with more segments to follow).

Report of Success

The TC-RESULT-L and TC-RESULT-NL primitives are used to indicate that an operation has been executed by the remote TC-USER.

- TC-RESULT-L

  Indicates the only or last segment of a result.

- TC-RESULT-NL

  Indicates a segment of a result (with more segments to follow).

Report of Failure

A TC-USER receiving an operation which it cannot execute, though it understands it, will issue a TC-U-ERROR request primitive, indicating the reason of the failure (Error parameter). The TC-USER which invoked the operation is informed by a TC-U-ERROR indication primitive.

Reject by TC-USER

A TC-USER rejects a component with the TC-U-REJECT request primitive, and is informed of rejection by the remote TC_User with the TC-U-REJECT indication primitive.

Cancel of an Operation: The TC-USER informs the local Component sub-layer of a cancel decision with the TC-U-CANCEL request primitive. No component is sent.

Reject of a Component by the Component Sub-Layer

While detecting that a received component is invalid, the Component sub-layer notifies the local TC-USER with the TC-L-REJECT indication primitive. The reject information is passed to the TC-USER and also retained in the Component sub-layer which uses it to form a Reject component.

The remote TC-USER is informed of the received Reject component through a TC-R-REJECT indication primitive.

Dialog Abort

TCAP may abort the association between users due to an abnormal situation. The structured dialog must then also be aborted. All associated operations are terminated and the TC-USERs are notified with the TC-P-ABORT indication primitive.

Closing a Transaction

A transaction is closed if TC-USER:

Issues:

- TC-END (ITU) or TC-RESPONSE (ANSI)
- TC-U-ABORT

Receives:

- TC-END (ITU) or TC-RESPONSE (ANSI)
- TC-U-ABORT
- TC-P-ABORT

Incoming Reject Components

An incoming reject component received with a TCAP Problem Code causes a TC-R-REJECT indication to be sent to the TC-USER. This indicates that the remote TCAP detected a problem with a previously sent component.

An incoming reject component received with a TC-USER Problem Code causes a TC-U-REJECT indication to be sent to the TC-USER. This indicates that the remote TC-USER detected a problem with a previously sent component and issued a TC-U-REJECT request.

A TC-L- REJECT indication is used when a local TCAP detects a problem with an incoming component. TCAP sends a TC-L- REJECT indication to the host and a Reject component to the remote TCAP.

When a component is rejected by TCAP, all subsequent components within the same TCAP message are discarded. Any components before the invalid component are sent to the host by the proper primitives.

Host Link Failure Handling

When host link failure is detected by the CSP:

- TCAP Restart procedures apply for all active subsystems, if the TCAP PPL Component TUSI Config Byte 3 is set as “1”.
- All local subsystems are set as Prohibited.

When the host link is recovered, it is the host’s responsibility to set all subsystems in service using the N_STATE primitive as allowed.

TCAP Restart

A system reset or switchover will result in a TCAP Restart. All active TCAP dialogs will return to the IDLE state. You can initiate a manual TCAP restart by sending a *PPL Event Request* message to the TCAP TUSI component with the event of TCAP Restart (0x1E).

Do not reuse the dialog ID immediately after a TCAP Restart as there may be some network messages remaining in queue for the old dialog.

PPL Config Byte 2 of the TCAP TUSI component indicates the Abort Reason sent in the TCAP Abort message following a TCAP Restart.

Notes on TCAP Primitives

For primitives in TCAP (e.g., for TCAP TUSI: TC-BEGIN, TC-CONTINUE, TC-UNI) the following applies:

- The maximum ICB length in the PPL Event Indication has a default value of 240.
- When any TC-Primitive exceeds 240, then a TCAP TUSI *PPL Event Indication* (event=0x1F) is sent to the host prior to the actual PPL Event Indication, which means that the following

actual PPL Event Indication is truncated. You can increase this value by using TCAP TUSI Config Bytes 4 and 5. See *SS7 PPL Information* for the config byte message description.

For the definition and usage of TCAP primitives, refer to the ITU White Book Q.711 “Functional Description of Transaction Capabilities” and Q.775 “Guideline for using TCAP.”

Supported Primitives

This section lists the SCCP and TCAP primitives supported by Dialogic’s SCCP/TCAP feature. Primitives are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in ICBs.

Important! SCCP/TCAP primitives should not be confused with Dialogic PPL primitives. TCAP Components should not be confused with Dialogic PPL components. See *PPL Information* for PPL event values and ICB formats.

The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive. See *SCCP/TCAP Parameter Information* for the data required for each parameter**.**

The ANSI specification does not define TCAP primitives to TC_User. Where common primitives are used, Dialogic followed the ITU Q.771 definitions.

ITU Primitives

The following ITU SCCP and TCAP primitives are supported by the *PPL Event Request* and *PPL Event Indication* messages. The primitive is supported by both messages unless noted otherwise.

TCAP

TC-BEGIN 0x01
TC-CONTINUE 0x02
TC-END 0x03
TC-UNI 0x04
TC-U-ABORT 0x05
TC-P-ABORT 0x06 (Indication only)
TC-RESULT-L 0x0C
TC-RESULT-NL 0x0D
TC-U-ERROR 0x0E
TC-L-CANCEL 0x0F (Indication only)
TC-U-REJECT (0x12)
TC-L-REJECT 0x13 (Indication only)
TC-R-REJECT 0x14 (Indication only)
TC-INVOKE 0x15
TC-U-CANCEL 0x16 (Request only)
TC-NOTICE 0x17 (Indication only)
TC_RESET_TIMER 0x19 (Request only)

SCCP

N-UNIT-DATA 0x01
N-NOTICE 0x02 (Indication only)
N-STATE 0x03
N-PC-STATE 0x04 (Indication only)

ANSI Primitives

The ANSI primitives are supported by both the *PPL Event Request* and *PPL Event Indication* message unless noted otherwise.

TCAP

TC-UNI 0x04
TC-U-ABORT 0x05
TC-P-ABORT 0x06 (Indication only)
TC-QUERY-WITH-PERMISSION 0x07
TC-QUERY-WITHOUT-PERMISSION 0x08
TC-CONVERSATION-WITH-PERMISSION 0x09
TC-CONVERSATION-WITHOUT-PERMISSION 0x0A
TC-RESPONSE 0x0B
TC-RESULT-L 0x0C
TC-RESULT-NL 0x0D
TC-U-ERROR 0x0E
TC-INVOKE-L 0x10
TC-INVOKE-NL 0x11
TC-U-REJECT 0x12
TC-L-REJECT 0x13 (Indication only)
TC-R-REJECT 0x14 (Indication only)
TC-U-CANCEL 0x16 (Request only)
TC-NOTICE 0x17 (Indication only)

SCCP

N-UNIT-DATA 0x01
N-NOTICE 0x02 (Indication only)
N-STATE 0x03
N-PC-STATE 0x04 (Indication only)

ITU TCAP Primitives

Overview This section lists the ITU TCAP primitives supported by the SCCP/ TCAP feature. Primitives are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in ICBs:

- SS7 TCAP Parameters (0x21).

Important! The TCAP ICB always includes a dialog ID which is four bytes.

ITU Primitives The following ITU TCAP primitives are supported by the *PPL Event Request* and *PPL Event Indication* messages. The primitive is supported by both messages unless noted otherwise.

- TC-UNI
- TC-BEGIN
- TC-CONTINUE
- TC-END
- TC-U-ABORT
- TC-P-ABORT (Indication only)
- TC-NOTICE (Indication only)
- TC-INVOKE
- TC-RESULT-L
- TC-RESULT-NL
- TC-U-ERROR
- TC-U-REJECT
- TC-L-CANCEL (Indication only)
- TC-U-CANCEL (Request only)
- TC-L-REJECT (Indication only)
- TC-R-REJECT (Indication only)
- N-UNIT-DATA
- N-NOTICE (Indication only)
- N-STATE
- N-PC-STATE (Indication only)
- TC_RESET_TIMER

ITU TCAP TUSI PPL Events

The TCAP User Interface is responsible for host API message validation. This section lists the PPL Events used to send ITU TCAP primitives in the *PPL Event Request* and *PPL Event Indication* messages. The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive. See *SCCP/TCAP Parameter Information* for the data required for each parameter.

The following events are used to send and receive ITU primitives for the PPL component, TCAP TUSI (0x70). The primitive IDs correspond to the PPL Event IDs for this component.

Note: An asterisk beside an event indicates that you should see** SCCP/TCAP Parameter Information **for the format.

0x01 TC-BEGIN

:

Request	Indication
TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) - MTP_DPC (0)	TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) - MTP_OPC (0)

0x02 TC-CONTINUE

Request	Indication
TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Quality of Service (O)	TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) - Component Present (M) SCCP Parameter ICB (O) - Quality of Service (O)

0x03 TC-END

Request	Indication
TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) - Termination Option (M) SCCP Parameter ICB (O) - Quality of Service (O)	TCAP Parameter ICB (M) - Application Context Name (O) - User Information (O) - Component Present (M SCCP Parameter ICB (O) - Quality of Service (O)

0x04 TC-UNI

This dialog primitive requests/indicates an Unstructured dialog. It corresponds to an Unidirectional TCAP package.

.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x05 TC-U-ABORT

This dialog primitive allows a TC-USER to terminate a dialog abruptly without transmitting any pending components.

Request	Indication
TCAP Parameter ICB (M) - User Abort Info (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)	TCAP Parameter ICB (M) - User Abort Info (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x06 TC-P-ABORT

This dialog primitive informs TC-USER that the dialog has been terminated by the TCAP because a protocol error has been detected. No pending components will be sent.

Request	Indication
	TCAP Parameter ICB (M) - P-Abort Cause (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x07 TC-QUERY-WITH-PERMISSION

This dialog primitive is similar to the ITU TC-BEGIN primitive. Under normal circumstances, it will cause a TCAP Query with Permission package to be sent to the network. An Indication indicates an incoming Query with Permission package, which starts a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) - MTP_DPC (0)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) - MTP_OPC (0)

0x08 TC-QUERY-WITHOUT-PERMISSION

This dialog primitive is the same as the TC-QUERY-WITH-PERMISSION except that it indicates to the peer that the transaction cannot be released. See the ANSI specification for more information (this primitive corresponds to the Query Without Permission package). This primitive starts a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) - MTP_DPC (0)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) - MTP_OPC (0)

0x09 TC-CONVERSATION-WITH-PERMISSION

This dialog primitive is similar to the ITU TC-CONTINUE. It corresponds to the TCAP Conversation package. It indicates the continuation of a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O) [1] - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [2] - Called Party Address (CDPA) or CDPA Elements (O) [2] - Quality of Service (O)
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2]CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0A TC-CONVERSATION-WITHOUT-PERMISSION

This primitive is similar to the TC-CONVERSATION-WITH-PERMISSION, except that the peer does not have permission. It corresponds to the ITU Conversation Without Permission package. It is the continuation of a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O) [1] - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [2] - Called Party Address (CDPA) or CDPA Elements (O) [2] - Quality of Service (O)
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2] CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0B TC-RESPONSE

This dialog primitive is similar to the ITU TC-END primitive. It corresponds to the ITU Response package when the termination option is set to Basic End. When the termination option is set to Pre-arranged End, this primitive ends the dialog locally.

Request	Indication
TCAP Parameter ICB (M) - Termination Option SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x0C TC-RESULT-L

This component primitive corresponds to the ITU Return Result Last Component. It is the only result or the last part of the segmented result of a successfully executed operation.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *

0x0D TC-RESULT-NL

This component primitive corresponds to the ITU Return Result Not Last Component. It is a non-final part of a segmented result of a successfully executed operation.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) * - Last Component (M)

0x0E TC-U-ERROR

This component primitive corresponds to the ITU Return Error component. It indicates that the operation failed.

.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Error Code (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Error Code (M) * - Parameter (O) * - Last Component (M)

0x10 TC-INVOKE-L

This component primitive corresponds to the ITU Invoke Last component. It is the only part or last part of the segmentation of the invocation of the operation. It may be the invocation of an operation or an invocation responding to another invoke.

Request	Indication
TCAP Parameter ICB (M) - Invoke ID (O) - Correlation ID (O) - Operation (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Invoke ID (O) - Correlation ID (O) - Operation (M) * - Parameter (O) * - Last Component (M)

0x11 TC-INVOKE-NL

This component primitive corresponds to the ITU Invoke Not LAst component. It is always an invocation responding to another invoke. It is a non-final segment of the invoke.

Request	Indication
TCAP Parameter ICB (M) - Invoke ID (M) - Correlation ID (M) - Operation (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Invoke ID (M) - Correlation ID (M) - Operation (M) * - Parameter (O) * - Last Component (M)

0x12 TC-U-REJECT

This component primitive corresponds to a Reject component. It is initiated because of a user decision of Reject an Incoming Component.

.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M) (see *SCCP/TCAP Parameter Information* for format) - Parameter (O) * - Last Component (M)

0x13 TC-L-REJECT

This component primitive indicates the detection of a protocol error in an incoming component. A Reject component is constructed and stored for this dialog, and is sent out upon the reception of another transaction layer primitive request.

.

Request	Indication
	TCAP Parameter ICB (M) - Invoke ID or Correlation ID (O) - Problem Code (M) (see *SCCP/TCAP Parameter Information* for format) - Last Component (M)

0x14 TC-R-REJECT

This indication informs the TC-USER that the remote TCAP rejected a previously sent component.

Request	Indication
	TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M) (see *SCCP/TCAP Parameter Information* for format) - Last Component (M)

0x16 TC-U-CANCEL

This is a user request to cancel an operation. No component is sent.

Request	Indication
TCAP Parameter ICB (M) - Invoke ID (M)	

0x17 TC-NOTICE

This primitive informs the TC-USER that the network service provider is unable to provide the requested service.

Request	Indication
	TCAP Parameter ICB (M) - Return Cause (M) Elements (0) - Called Party Address (CDPA) or CDPA

0x19 TC-RESET TIMER

This primitive resets the Invoke timer for TC_INVOKE primitive initiated from the CSP.

Request	Indication
Dialog ID (M) Invoke ID (M)	

ANSI TCAP Primitives

Overview This section lists the ANSI TCAP primitives supported by the SCCP/TCAP feature. Primitives are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in ICBs:

- SS7 TCAP Parameters (0x21).

Important! The TCAP ICB always includes a dialog ID which is four bytes.

ANSI Primitives The ANSI primitives are supported by both the *PPL Event Request* and *PPL Event Indication* message unless noted otherwise.

- TC-UNI
- TC-QUERY-WITH-PERMISSION
- TC-QUERY-WITHOUT-PERMISSION
- TC-CONVERSATION-WITH-PERMISSION
- TC-CONVERSATION-WITHOUT-PERMISSION
- TC-RESPONSE
- TC-U-ABORT
- TC-P-ABORT (Indication only)
- TC-NOTICE (Indication only)
- TC-INVOKE-L
- TC-INVOKE-NL
- TC-RESULT-L
- TC-RESULT-NL
- TC-U-ERROR
- TC-U-REJECT
- TC-L-REJECT (Indication only)
- TC-R-REJECT (Indication only)
- TC-U-Cancel (Request only)
- N-UNIT-DATA
- N-NOTICE (Indication only)
- N-STATE

- N-PC-STATE (Indication only)

ANSI TCAP TUSI PPL Events

The TCAP User Interface is responsible for host API message validation. This section lists the PPL Events used to send ANSI TCAP primitives in the *PPL Event Request* and *PPL Event Indication* messages.The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive. See *SCCP/TCAP Parameter Information* for the data required for each parameter.

The following events are used to send and receive ANSI primitives for the PPL component, TCAP TUSI (0x70). The primitive IDs correspond to the PPL Event IDs for this component. The ANSI specification does not define TCAP primitives to TC_User. Where common primitives are used, Dialogic followed the ITU Q.771 definitions.

Note: An asterisk beside an event indicates that you should see** SCCP/TCAP Parameter Information **for the format.

0x04 TC-UNI

This dialog primitive requests/indicates an Unstructured dialog. It corresponds to an Unidirectional TCAP package. Note that the TCAP ICB always includes a dialog ID.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x05 TC-U-ABORT

This dialog primitive allows a TC-USER to terminate a dialog abruptly without transmitting any pending components.

Request	Indication
TCAP Parameter ICB (M) - User Abort Info (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)	TCAP Parameter ICB (M) - User Abort Info (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x06 TC-P-ABORT

This dialog primitive informs TC-USER that the dialog has been terminated by the TCAP because a protocol error has been detected. No pending components will be sent.

Request	Indication
	TCAP Parameter ICB (M) - P-Abort Cause (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x07 TC-QUERY-WITH-PERMISSION

This dialog primitive is similar to the ITU TC-BEGIN primitive. Under normal circumstances, it will cause a TCAP Query with Permission package to be sent to the network. An Indication indicates an incoming Query with Permission package, which starts a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x08 TC-QUERY-WITHOUT-PERMISSION

This dialog primitive is the same as the TC-QUERY-WITH-PERMISSION except that it indicates to the peer that the transaction cannot be released. See the ANSI specification for more information (this primitive corresponds to the Query Without Permission package). This primitive starts a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x09 TC-CONVERSATION-WITH-PERMISSION

This dialog primitive is similar to the ITU TC-CONTINUE. It corresponds to the TCAP Conversation package. It indicates the continuation of a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O)[1] - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [2] - Called Party Address (CDPA) or CDPA Elements (O) [2] - Quality of Service (O)
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2] CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0A TC-CONVERSATION-WITHOUT-PERMISSION

This primitive is similar to the TC-CONVERSATION-WITH-PERMISSION, except that the peer does not have permission. It corresponds to the ITU Conversation Without Permission package. It is the continuation of a dialog.

Request	Indication
TCAP Parameter ICB (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O)[1] - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O)[2] - Called Party Address (CDPA) or CDPA Elements (O)[2] - Quality of Service (O)
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2] CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0B TC-RESPONSE

This dialog primitive is similar to the ITU TC-END primitive. It corresponds to the ITU Response package when the termination option is set to Basic End. When the termination option is set to Pre-arranged End, this primitive ends the dialog locally.

Request	Indication
TCAP Parameter ICB (M) - Termination Option SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)	TCAP Parameter ICB (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O)

0x0C TC-RESULT-L

This component primitive corresponds to the ITU Return Result Last Component. It is the only result or the last part of the segmented result of a successfully executed operation.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *

0x0D TC-RESULT-NL

This component primitive corresponds to the ITU Return Result Not Last Component. It is a non-final part of a segmented result of a successfully executed operation.

.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Parameter (O) * - Last Component (M)

0x0E TC-U-ERROR

This component primitive corresponds to the ITU Return Error component. It indicates that the operation failed.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Error Code (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Error Code (M) * - Parameter (O) * - Last Component (M)

0x10 TC-INVOKE-L

This component primitive corresponds to the ITU Invoke Last component. It is the only part or last part of the segmentation of the invocation of the operation. It may be the invocation of an operation or an invocation responding to another invoke.

Request	Indication
TCAP Parameter ICB (M) - Invoke ID (O) - Correlation ID (O) - Operation (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Invoke ID (O) - Correlation ID (O) - Operation (M) * - Parameter (O) * - Last Component (M)

0x11 TC-INVOKE-NL

This component primitive corresponds to the ITU Invoke Not Last component. It is always an invocation responding to another invoke. It is a non-final segment of the invoke.

Request	Indication
TCAP Parameter ICB (M) - Invoke ID (M) - Correlation ID (M) - Operation (M) * - Parameter (O) *	TCAP Parameter ICB (M) - Invoke ID (M) - Correlation ID (M) - Operation (M) * - Parameter (O) * - Last Component (M)

0x12 TC-U-REJECT

This component primitive corresponds to a Reject component. It is initiated because of a user decision of Reject an Incoming Component.

Request	Indication
TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M) - Parameter (O) *	TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M)(see *SCCP/TCAP Parameter Information* for format) - Parameter (O) * - Last Component (M)

0x13 TC-L-REJECT

This component primitive indicates the detection of a protocol error in an incoming component. A Reject component is constructed and stored for this dialog, and is sent out upon the reception of another transaction layer primitive request.

Request	Indication
	TCAP Parameter ICB (M) - Invoke ID or Correlation ID (O) - Problem Code (M) (see *SCCP/TCAP Parameter Information* for format) - Last Component (M)

0x14 TC-R-REJECT

This indication informs the TC-USER that the remote TCAP rejected a previously sent component.

Request	Indication
	TCAP Parameter ICB (M) - Correlation ID (O) - Problem Code (M) (see *SCCP/TCAP Parameter Information* for format) - Last Component (M)

ITU and ANSI SCCP Primitives

Overview

This section lists the SCCP primitives supported by the SCCP/TCAP feature. Primitives are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in the ICB:

- SS7 SCCP Parameters (0x20).

Important! SCCP/TCAP primitives should not be confused with Dialogic PPL primitives. SCCP Components should not be confused with Dialogic PPL components. See the *PPL Information* for PPL event values and the *API Reference* ICB formats.

The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive. See *SCCP/TCAP Parameter Information* for the data required for each parameter.

ITU/ANSI SCCP Events

The SCCP User Interface is responsible for message validation and format conversion. The following events are used to send and receive primitives for the PPL component, SCCP SUSI (0x67).

N-UNIT-DATA

Request	Indication
SCCP Parameter ICB (M) - User Data (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O)	Same as Request.

N-NOTICE (0x22)

Request	Indication
	SCCP Parameter ICB (M) - User Data (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Reason for Return (M)

N-STATE (0x03)

Request	Indication
SCCP Parameter ICB (M) - Subsystem Number (M) - Subsystem Status (M)	SCCP Parameter ICB (M) - Destination Point Code (DPC) (M) - DPC Status (O) - Remote SCCP Status (O)

Important! When you use the N_STATE event to set a subsystem prohibit, all of the active TCAP dialogs associated with the SSN are automatically released.

N-PCSTATE (0x04)

Request	Indication
	SCCP Parameter ICB (M) - Destination Point Code (DPC) (M) - DPC Status (O) - Remote SCCP Status (O)

TCAP Primitive Set Interface

Purpose

The TCAP Primitive Set Interface feature is based on the existing TCAP software architecture. TCAP and the SS7 layer below it reside on the CSP and the TC-User resides on the host.

TCAP communicates with the TC-User through multiple TC-primitives so that a dialog can be embedded into a single *PPL Event Request* or *PPL Event Indications* API message.

This feature provides the following:

- Reduces the API level transport and processing overhead on various software components, such as the SS7 host communication and transport layer, and host application message processing and transport layer.
- Reduces message traffic on the internal Ethernet or HDLC depending on the architecture, thereby improving overall system performance.

TCAP Primitive Set Interface Description

The host application sends and receives messages to and from the TCAP USer Interface (TUSI) PPL component for the primitive set interface. Each TC-primitive type is uniquely identified by a PPL Event ID in the *PPL Event Request* and *PPL Event Indication* API messages. These API messages include all of the ICBs for each individual TC-primitive. A component TC-primitive includes a TCAP parameter ICB. A dialog TC-primitive includes a TCAP parameter ICB plus an optional SCCP parameter ICB for SCCP addresses.

TCAP Outgoing Message

The TUSI parses and validates the *PPL Event Request API* message with TC primitive set PPL Event ID from the TC-User. After validation, the TUSI sends the component primitive data and the dialog primitive data to the TCAP PPL components according to the TCAP protocol.

- The TUSI first sends each component primitive data to the TCAP PPL components in the order presented in the API message.
- The TUSI then sends the dialog primitive data to the TCAP PPL components and sends a Positive Acknowledgement to the TC-User.

- Other TCAP PPL components will process each of these primitives individually.

TCAP Incoming Message

When TCAP receives an incoming message with one or more components, the TCAP PPL components process this message according to the TCAP protocol.

- After processing, the dialog primitive data first and then the component primitive data are sent to the TUSI.
- The TUSI packs the dialog primitive ICB(s), followed by all the component ICBs in the order it receives them.
- It then sends all of these ICBs in one *PPL Event Indication* API message to the TC-User.

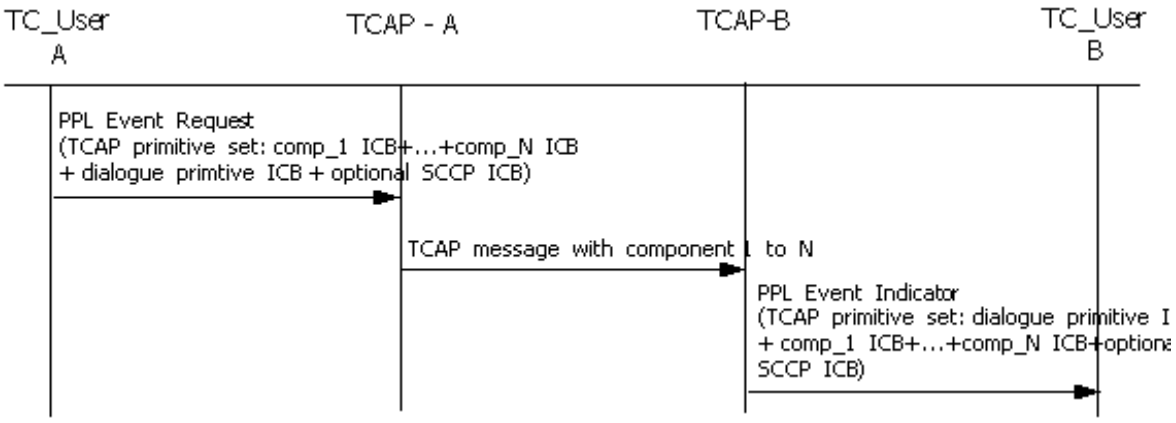
TUSI Negative Acknowledgements (NACK)

The TUSI sends a negative acknowledgement (NACK) to the TC-User if it detects syntax errors in the *PPL Event Request* API message. The TC-User may re-send the API message after receiving a NACK from TCAP.

> **Important!** The TC-User is responsible for the correctness and consistency of the contents in the *PPL Event Request* API message. After the TUSI validates the API message syntax and sends the dialog and component primitives to other TCAP PPL components, TCAP would consider any detected error as an unrecoverable error for the dialog primitives, therefore the dialog primitives will be aborted in this case. The TC-User as well as the network (if needed) will be informed.

TCAP Primitive Set Interface Call Flow

The following call flow provides an example of the TCAP to TC-User interface using the TCAP Primitive Set Interface feature.



TCAP Primitive Sets PPL Events

The TCAP Primitive Sets PPL Events listed below are provided for the TCAP Primitive Set Interface. One PPL Event is defined for each dialog TC-primitive type that is allowed to have attached component(s). The PPL Events listed below are in addition to the currently supported PPL Events for individual TC-primitives.

ANSI Variant TC Primitive Set PPL Events

TC_UNI Primitive Set (0x31) *(PPL Event Indication and PPL Event Request)*

TC_QUERY_WITH_PERMISSION Primitive Set (0x35) *(PPL Event Indication and PPL Event Request)*

TC_QUERY_WITHOUT_PERMISSION Primitive Set (0x36) *(PPL Event Indication and PPL Event Request)*

TC_CONVERSATION_WITH_PERMISSION Primitive Set (0x37) *(PPL Event Indication and PPL Event Request)*

TC_CONVERSATION_WITHOUT_PERMISSION Primitive Set (0x38) *(PPL Event Indication and PPL Event Request)*

TC_RESPONSE Primitive Set (0x39) *(PPL Event Indication and PPL Event Request)*

ITU/ETSI Variant TC Primitive Set PPL Events

TC_UNI Primitive Set (0x31) *(PPL Event Indication and PPL Event Request)*

TC_BEGIN Primitive Set (0x32) *(PPL Event Indication and PPL Event Request)*

TC_CONTINUE Primitive Set (0x33) *(PPL Event Indication and PPL Event Request)*

TC_END Primitive Set (0x34) *(PPL Event Indication and PPL Event Request)*

ANSI/ITU/ETSI Variant TC Primitive PPL Event

PPL_EVENT_TCAP_INITIATED_U_ABORT (0x21) *(PPL Event Indication)*

This PPL Event Indication is initiated by TCAP due to software internal inconsistency and unrecoverable errors. These errors can be caused by incorrect data passed from the TC-User or other software module in the system as well as TCAP internal software errors.

> **Important!** This event is different from TC_U_ABORT event that indicates the remote TCAP User aborted the transaction.

TC Primitive Set

To align with the ICBs in individual TC-primitives, a TC Primitive Set includes a TCAP dialog primitive ICB and zero, one, or more TCAP component ICBs, and an optional SCCP ICB for SCCP addresses.

A zero-component is allowed to be compliant with the TCAP standard where the component portion is optional in some TCAP messages. Whether it is mandatory for each TCAP message to include a component is based on the specific TCAP standard variants.

When using the TCAP Primitive Set Interface, use the following TC-primitive variants in addition to the TC primitive set events:

ANSI Variants

TC_U_ABORT *(PPL Event Indication and PPL Event Request)*

TC_P_ABORT *(PPL Event Indication)*

TC_NOTICE *(PPL Event Indication)*

ITU/ETSI Variants

TC_U_ABORT *(PPL Event Indication and PPL Event Request)*

TC_P_ABORT *(PPL Event Indication)*

TC_NOTICE *(PPL Event Indication)*

TC_L_CANCEL *(PPL Event Indication)*

TCAP Primitive Options

TCAP primitive configuration options, per SS7 stack, are provided to select either the Individual TCAP Primitive API option or the TCAP Primitive Set Interface option.

The default is the Individual TC Primitive API option for backward compatibility considerations. The TCAP PPL Component TUSI Configuration Byte is used for this purpose.

Use the TCAP primitive ICB subtype for TCAP primitive sets.

ICB Subtype - TCAP Primitive ICB

The TCAP primitive ICB subtype (0x65) has two additional bytes for the TCAP TC-primitive ID in the following ICB formats:

- Data (bytes 8 and 9)
- Extended Data (bytes 10 and 11)

Compared to the existing TCAP parameter ICB, the TCAP TC-primitive ID value is same as the corresponding PPL Event ID value. The TCAP parameter TLVs for each TC primitive remain same as in current TCAP parameter ICB.

TCAP TUSI PPL Configuration Byte

Byte	Description	Value
0x09	Selects TCAP to TC-User interface format	0x00 = Use individual TC-Primitive (Default) 0x01 = Use TCAP Primitive Set Interface

Examples

You can view TCAP Primitive Set API messaging examples in *Example Configurations and TCAP API Messaging*.

ITU TCAP Primitive Sets

Overview

The TCAP Primitive Set feature provides an interface to combine one TCAP dialog, TC-Primitive, and its multiple associated components into one *PPL Event Request* or *PPL Event Indication*. This section lists the ITU TCAP Primitive Sets supported by Dialogic's SCCP/TCAP feature. Primitive Sets are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in ICBs:

- SS7 TCAP Primitive ICB (0x65)
- SS7 TCAP Parameters ICB (0x21)

Important! The maximum number of ICBs for the TCAP primitive set is 16. If this maximum is exceeded, the CSP sends negative acknowledgement 0x5401.

The TCAP parameter ICBs always include a dialog ID which is four bytes. The TCAP Primitive ICB includes a four-byte dialog ID and a two-byte Event ID.

ITU Primitive Sets

The following ITU TCAP Primitive Sets are supported by the *PPL Event Request* and *PPL Event Indication* messages. Each primitive set is supported by both messages unless noted otherwise.

- TC-BEGIN Primitive Set
- TC-CONTINUE Primitive Set
- TC-END Primitive Set
- TC-UNI Primitive Set

Needed Primitives

When using the Primitive Set interface you may also need the following primitives.

- TC-U-ABORT (Request and Indication)
- TC-P-ABORT (Indication only)
- TC-NOTICE (Indication only)
- TC-L-CANCEL (Indication only)

ITU TCAP TUSI PPL Events

The TCAP User Interface is responsible for host API message validation. This section lists the PPL Events used to send ITU TCAP primitive sets in the *PPL Event Request* and *PPL Event Indication* messages. The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive set. See *SCCP/TCAP Parameter Information* for the data required for each parameter.

The following events are used to send and receive ITU primitive sets for the PPL component, TCAP TUSI (0x70). The primitive set IDs correspond to the PPL Event IDs for this component.

In the tables below you will see: TCAP Primitive ICB for a component (none or multiple)*. This refers to the Primitive ID in the *SS7 TCAP Primitive ICB 0x65*. The Primitive ID in this ICB has the same ID as the Event ID in the TCAP Primitive Interface for the same component. The Primitive IDs used for the following TCAP Primitive Sets are:

TC-RESULT-L 0x0C
TC-RESULT-NL 0x0D
TC-U-ERROR 0x0E
TC-U-REJECT (0x12)
TC-L-REJECT 0x13 (Indication only)
TC-R-REJECT 0x14 (Indication only)
TC-INVOKE 0x15
TC-U-CANCEL 0x16 (Request only)

The Parameter TLV for a particular Primitive ID remains the same as if they were in the TCAP Primitive Interface for the same Event ID.

0x01 TC-BEGIN Primitive Set

:

Request	Indication
TCAP Primitive ICB0x65 (M) - Application Context Name (O) - User Information (O) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) - MTP_DPC (O) TCAP Primitive ICB for a component (none or multiple)*	TCAP Primitive ICB 0x65 (M) - Application Context Name (O) - User Information (O) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) - MTP_OPC (O) TCAP Primitive ICB for a component (none or multiple)*

0x02 TC-CONTINUE Primitive Set

Request	Indication
TCAP Primitive ICB 0x65 (M) - Application Context Name (O) - User Information (O) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*	TCAP Primitive ICB 0x65 (M) - Application Context Name (O) - User Information (O) - Component Present (M) SCCP Parameter ICB (O) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*

0x03 TC-END Primitive Set

Request	Indication
TCAP Primitive ICB 0x65 (M) - Application Context Name (O) - User Information (O) - Termination Option (M) SCCP Parameter ICB (O) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*	TCAP Primitive ICB 0x65 (M) - Application Context Name (O) - User Information (O) - Component Present (M) SCCP Parameter ICB (O) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*

0x04 TC-UNI Primitive Set

This dialog primitive set requests/indicates an Unstructured dialog. It corresponds to an Unidirectional TCAP package.

.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (none or multiple)*

ANSI TCAP Primitive Sets

Overview

The TCAP Primitive Set feature provides an interface to combine one TCAP dialog, TC-Primitive, and its multiple associated components into one *PPL Event Request* or *PPL Event Indication*. This section lists the ANSI TCAP Primitive Sets supported by Dialogic's SCCP/TCAP feature. Primitive Sets are sent to and received from the CSP in the *PPL Event Request* and *PPL Event Indication* messages in ICBs:

- SS7 TCAP Primitives ICB (0x65)
- SS7 TCAP Parameters ICB (0x21)

Important! The maximum number of ICBs for the TCAP primitive set is 16. If this maximum is exceeded, the CSP sends negative acknowledgement 0x5401.

The TCAP ICBs always includes a dialog ID which is four bytes.The TCAP Primitive ICB includes a four-byte dialog ID and a two-byte Event ID.

ANSI Primitives

The ANSI Primitive Sets are supported by both the *PPL Event Request* and *PPL Event Indication* message unless noted otherwise.

- TC-UNI Primitive Set
- TC-QUERY-WITH-PERMISSION Primitive Set
- TC-QUERY-WITHOUT-PERMISSION Primitive Set
- TC-CONVERSATION-WITH-PERMISSION Primitive Set
- TC-CONVERSATION-WITHOUT-PERMISSION Primitive Set
- TC-RESPONSE Primitive Set

Needed Primitives

When using the Primitive Set interface you may also need the following primitives.

- TC-U-ABORT (Request and Indication)
- TC-P-ABORT (Indication only)
- TC-NOTICE (Indication only)

ANSI TCAP TUSI PPL Events

The TCAP User Interface is responsible for host API message validation. This section lists the PPL Events used to send ANSI TCAP primitives in the *PPL Event Request* and *PPL Event Indication*

messages.The required ICBs with mandatory (M) and optional (O) parameters are shown with each primitive. See *SCCP/TCAP Parameter Information* for the data required for each parameter.

The following events are used to send and receive ANSI primitives for the PPL component, TCAP TUSI (0x70). The primitive IDs correspond to the PPL Event IDs for this component. The ANSI specification does not define TCAP primitives to TC_User. Where common primitives are used, Dialogic followed the ITU Q.771 definitions.

In the tables below you will see: TCAP Primitive ICB for a component (one or more)*. This refers to the Primitive ID in the *SS7 TCAP Primitive ICB 0x65*. The Primitive ID in this ICB has the same ID as the Event ID in the TCAP Primitive Interface for the same component. The Primitive IDs used for the following TCAP Primitive Sets are:

TC-RESULT-L 0x0C
TC-RESULT-NL 0x0D
TC-U-ERROR 0x0E
TC-INVOKE-L 0x10
TC-INVOKE-NL 0x11
TC-U-REJECT 0x12
TC-L-REJECT 0x13 (Indication only)
TC-R-REJECT 0x14 (Indication only)
TC-U-CANCEL 0x16 (Request only)

The Parameter TLV for a particular Primitive ID remains the same as if they were in the TCAP Primitive Interface for the same Event ID.

0x04 TC-UNI Primitive Set

This dialog primitive set requests/indicates an Unstructured dialog. It corresponds to an Unidirectional TCAP package. Note that the TCAP ICB always includes a dialog ID.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*

0x07 TC-QUERY-WITH-PERMISSION Primitive Set

This dialog primitive set is similar to the ITU TC-BEGIN primitive. Under normal circumstances, it will cause a TCAP Query with Permission package to be sent to the network. An Indication indicates an incoming Query with Permission package, which starts a dialog.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*

0x08 TC-QUERY-WITHOUT-PERMISSION Primitive Set

This dialog primitive set is the same as the TC-QUERY-WITH-PERMISSION except that it indicates to the peer that the transaction cannot be released. See the ANSI specification for more information (this primitive corresponds to the Query Without Permission package). This primitive starts a dialog.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (M) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (M) - Calling Party Address (CGPA) or CGPA Elements (M) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*

0x09 TC-CONVERSATION-WITH-PERMISSION Primitive Set

This dialog primitive set is similar to the ITU TC-CONTINUE. It corresponds to the TCAP Conversation package. It indicates the continuation of a dialog.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O)[1] - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [2] - Called Party Address (CDPA) or CDPA Elements (O) [2] - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2] CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0A TC-CONVERSATION-WITHOUT-PERMISSION Primitive Set

This primitive set is similar to the TC-CONVERSATION-WITH-PERMISSION, except that the peer does not have permission. It corresponds to the ITU Conversation Without Permission package. It is the continuation of a dialog.

Request	Indication
TCAP Primitive ICB 0x65 (M) - No parameters included SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) [1] - Called Party Address (CDPA) or CDPA Elements (O)[1] - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O)[2] - Called Party Address (CDPA) or CDPA Elements (O)[2] - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*
[1] CGPA and CDPA are stored for the dialog when it is initialed. TC-USER provides CGPA and CDPA only if they are changed. [2] CGPA and CDPA are sent to the host only if they differ from the stored values.	

0x0B TC-RESPONSE Primitive Set

This dialog primitive set is similar to the ITU TC-END primitive. It corresponds to the ITU Response package when the termination option is set to Basic End. When the termination option is set to Pre-arranged End, this primitive ends the dialog locally.

Request	Indication
TCAP Primitive ICB 0x65 (M) - Termination Option SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*	TCAP Primitive ICB 0x65 (M) - Component Present (M) SCCP Parameter ICB (O) - Calling Party Address (CGPA) or CGPA Elements (O) - Called Party Address (CDPA) or CDPA Elements (O) - Quality of Service (O) TCAP Primitive ICB for a component (one or more)*

SCCP/TCAP Parameter Information

Overview

This section contains information on SCCP and TCAP parameters inserted into the SCCP and TCAP ICBs. Note special instructions for the parameters described below.

Description of Parameters

Called Party Address (CDPA)

The CDPA in the primitive request is used if specified. If DPC is not included in the CDPA, the MTP DPC or adjacent translator must be provided.

The following sequence is used for deriving the DPC:

- Use DPC in CDPA if provided; otherwise,
- If CDPA routes on DPC/SSN, then use the MTP DPC in the primitive or in the SSN default table. If CDPA routes on Global Title, then use the first adjacent translator as the DPC.

Important! As an SSP, Global Title Translation (GTT) is not provided.

Calling Party Address (CGPA)

The CGPA in the primitive request is used if provided; otherwise,

- If a TCAP primitive, the CGPA stored for this dialog is used.
- If configured for this SSN, the CGPA in the SSN default table is used.

Component Present

This indicates whether or not components are present. If components are present, then only syntactically valid and opportune components are delivered to the destination TC-USER.

Dialog ID

This parameter appears in the dialog handling and component handling primitives. to associate components with a dialog. The same dialog ID must be used within the same dialog. In a Unidirectional primitive, the same dialog ID assures that all components with the identical dialog ID are blocked together in the same Unidirectional message destined for the same destination address.

For structured dialogs, the dialog ID is used to identify all components belonging to the same dialog from the beginning of the dialog to its end. The dialog ID maps onto the transaction IDs exchanged in the messages between a pair of nodes.

Error Code

Use the NATIONAL-ERR-CODE parameter type (0x47) if a national error code is used, or the PRIVATE-ERR-CODE tag (0x48) if a private error code is used.

Last Component

Used in primitives of the Indication type to designate the last component of a message. Note that indication of the last part of the result of an operation is via the name of the primitive.

In the case of multiple components received within a TCAP message, a TC-transaction indication primitive followed by several TC-component indication primitives are sent to the TC-USER. The Last Component parameter in the TC-component primitive specifies if this component is the last component in the message.

This should not be confused with the Last/Not Last primitive type, which indicates the last or not last part of an operation or result of an operation.

Operation

Use the GLOBAL-OPER-CODE parameter type (0x2C) if a national operation code is used, or a LOCAL-OPER-CODE tag (0x2D) if a private operation code is used.

Parameter (ITU only)

Use the PARAMETER-SEQ parameter type (0x4A) to send a sequence of parameters or the PARAMETER-SET tag (0x4B) to send a parameter set.

Problem Code (ITU only)

Use PROBLEM-TYPE-CODE parameter type.

Quality of Service

The QOS in the primitive request is used if specified, otherwise:

- The QOS in the SSN default table is used if it is configured for the subsystem, otherwise;
- The following default configuration is used:

  Return Option = Do Not Return on Error

  Sequence Option = Sequence Not Guaranteed

  Message Priority (ANSI)=0x00

Termination

Indicates which scenario is chosen by the TC-USER for the end of the dialog (prearranged or basic).

Timeout (ITU only)

The timeout value is in 10 ms units.

ITU Parameter Information

This section contains information on ITU SCCP and TCAP parameters inserted into the SCCP and TCAP ICBs.

ITU SCCP Parameter IDs

0x66	CGPA (Calling Party Address)
0x67	CGPA (Calling Party Address) Element
0x68	CDPA (Called Party Address)
0x69	CDPA (Called Party Address) Element
0x6A	QOS (Quality of Service)
0x6B	User Data
0x6C	CDPA (Called Party Address) DPC
0x6D	Return Reason
0x6E	SSN (Subsystem Number)
0x6F	SPC (Signaling Point Code)
0x70	SSN (Subsystem Number) Status
0x71	SCCP Signaling Point Status
0x72	Remote SCCP Status
0x73	MTP OPC
0x74	MTP DPC

ITU TCAP Parameter IDs

0x0F	Abort Reason
0x11	P-Abort Cause
0x1F	Invoke ID
0x20	Link ID
0x28	Invoke Class
0x29	Invoke Timeout
0x2A	Problem Type Code
0x2E	Parameter
0x2C	Global Operation Code
0x2D	Local Operation Code
0x30	Last Component
0x37	Protocol Version
0x38	Application Context Name
0x39	User Info
0x3A	Termination Option
0x43	Dialog_AS_ID
0x44	Unidialog_AS_ID
0x45	Component Present
0x47	Global Error Code
0x48	Local Error Code
0x6D	Return Reason

ANSI Parameter Information

This section contains information on ANSI SCCP and TCAP parameters inserted into the SCCP and TCAP ICBs.

ANSI SCCP Parameter IDs

0x66	CGPA (Calling Party Address)
0x67	CGPA (Calling Party Address) Element

0x68	CDPA (Called Party Address)
0x69	CDPA (Called Party Address) Element
0x6A	QOS (Quality of Service)
0x6B	User Data
0x6C	CDPA (Called Party Address) DPC
0x6D	Return Reason
0x6E	SSN (Subsystem Number)
0x6F	SPC (Signaling Point Code)
0x70	SSN (Subsystem Number) Status
0x71	SCCP Signaling Point Status
0x73	MTP OPC
0x74	MTP DPC

ANSI TCAP Parameter IDs

0x0F	User Abort Info
0x11P	Abort Cause
0x1F	Invoke ID
0x20	Correlation ID
0x2A	Problem Type Code
0x2C	National Operation Code
0x2D	Local Operation Code
0x30	Last Component
0x3A	Termination Option
0x45	Component Present
0x47	National Error Code
0x48	Private Error Code
0x4A	Parameter Sequence
0x4B	Parameter Set
0x6D	Return Reason

Parameter Format The following table illustrates the parameter format for parameters inserted into the SCCP and TCAP ICBs.

Byte	Field
Data[0]	Parameter 1 ID
Data[1]	
Data[2]	Parameter Length
Data[3]	
Data[4+]	Parameter-specific Fields

Parameter Data This section shows the specific data for each parameter.

Abort Reason

Byte	Value(s)
Data[4] Reason	0x00 = User-defined 0x01 = Application Context Name Not Supported (ITU Only) 0x02 - Dialog Refused

Calling Party Address (CGPA)

Byte	Value(s)
Data[4+] Calling Party Address	Conforms to ANSI or ITU specification.

Calling Party Address (CGPA) Element

Byte	Value(s)
Data[4] Routing Indication	0x00 = Route on GT 0x01 = Route on DPC/SSN
Data[5] National/International Flag	0x00 = Coded International 0x01 = Coded National 0x02 = Coded according to China Specification (same as International except the point codes are 24 bits)

Byte	Value(s)
Data[6] Subsystem Number	
Data[7–10] Point Code	
Data[11] Global Title Indicator	
Data[12] Global Title Length	Variable
:	:
Data[N] Global Title	

Note:

1. Subsystem Number is **`0x00`** if unknown or not provided.
2. The Point Code is **`0xFFFFFFFF`** if unknown or not provided.
3. The Calling Party Address is constructed with the elements provided and converted to ANSI or ITU format according to the appropriate specification.
4. The National/International flag applies only to indication. In a request, SCCP codes the CGPA according to the SCCP SCLC Component SCLC Configuration Byte 1.
5. “Global Title Indicator” in Data[11] corresponds to the “Global Title Indicator” in the SCCP message CGPA.

Called Party Address (CDPA) Element

Byte	Value(s)
Data[4] Routing Indication	0x00 = Route on GT 0x01 = Route on DPC/SSN
Data[5] National/International Flag	0x00 = Coded International 0x01 = Coded National 0x02 = Coded according to China Specification (same as International except the point codes are 24 bits)
Data[6] Subsystem Number	
Data[7–10] Point Code	
Data[11] Global Title Indicator	
Data[12] Global Title Length	(Variable)
:	:
Data[N] Global Title	

Note

1. Subsystem Number is **`0x00`** if unknown or not provided.
2. The Point Code is **`0xFFFFFFFF`** if unknown or not provided.
3. The Called Party Address is constructed with the elements provided and converted to ANSI or ITU format according to the appropriate specification.
4. The National/International flag applies only to indication. In a request, SCCP codes the CDPA according to the SCCP PPL SCLC Component Config Byte 1.
5. “Global Title Indicator” in Data[11] corresponds to the “Global Title Indicator” of the SCCP message CDPA.

Class

Byte	Value(s)
Data[4] Class	Class Number (1–4)

Component Present

Byte	Value(s)
Data[4]	0x00 Not Present 0x01 Present

Dialog_AS_ID

The values for the Dialog_AS_ID parameter are automatically stored in the CSP, consistent with ITU TCAP specifications. They cannot be modified by the host if you use the ITU White Book.

Byte	Value(s)
Data[4] Status	0x01 = Signaling Point Inaccessible 0x02 = SPC Congested 0x03 = SPC Accessible

Error Code

Byte	Value(s)
Data[4] Code	Global or Local Error Code

Last Component

Byte	Value(s)
Data[4]	0x00 Last Component 0x01 Not Last Component

Operation

Byte	Value(s)
Data[4] Code	Global or Local Operation Code

P Abort Cause (ITU)

Byte	Value(s)
Data[4] Cause	0x00 = Unrecognized Message Type 0x01 = Unrecognized Transaction ID 0x02 = Badly Formatted Transaction Portion 0x03 = Incorrect Transaction Portion 0x04 = Resource Limitation 0x05 = Abnormal dialog 0x06 = No Common dialog Portion

P Abort Cause (ANSI)

Byte	Value(s)
Data[4] Cause	0x01 = Unrecognized Package Type 0x02 = Incorrect Transaction Portion 0x03 = Badly Structured Transaction Portion 0x04 = Unrecognized Transaction ID 0x05 = Permission to Release Problem 0x06 = Resource Unavailable 0x07 = Unrecognized Dialog Portion ID 0x08 = Bad Structured Dialog Portion 0x09 = Missing Dialog Portion 0x0A = Inconsistent Dialog Portion

Problem Type Code (ITU)

Byte	Value(s)
Data[4] Problem Type	0x01 = General 0x02 = Invoke 0x03 = Return Result 0x04 = Return Error
Data[5] Problem Code	General 0x00 = Unrecognized Component 0x01 = Mis-typed Component 0x02 = Badly Structured Component Invoke 0x00 = Duplicate Invoke ID 0x01 = Unrecognized Operation 0x02 = Mis-typed Parameter 0x03 = Resource Limitation 0x04 = Initiating Release 0x05 = Unrecognized Link ID 0x06 = Linked Response Unexpected 0x07 = Unexpected Linked Operation Return Result 0x00 = Unrecognized Invoke ID 0x01 = Return Result Unexpected 0x02 = Mis-typed Parameter Return Error 0x00 = Unrecognized Invoke ID 0x01 = Return Error Unexpected 0x02 = Unrecognized Error 0x03 = Unexpected Error 0x04 = Mis-typed Parameter

Problem Type Code (ANSI)

Byte	Value(s)
Data[4] Problem Type	0x01=General 0x02=Invoke 0x03=Return Result 0x04=Return Error 0x05=Transaction Portion

Byte	Value(s)
Data[5] Problem Code	**All Types** 0x00=Not Used 0xFF=Reserved **General** 0x01=Unrecognized Component Type 0x02=Incorrect Component Portion 0x03=Badly Structured Component Portion **Invoke** 0x01=Duplicate Invoke ID 0x02=Unrecognized Operation Code 0x03- Incorrect Parameter 0x04=Unrecognized Correlation ID 0x0F=Resource Unavailable **Return Result** 0x01=Unrecognized Correlation ID 0x02=Unexpected Return Result 0x03=Incorrect Parameter **Return Error** 0x01=Unrecognized Correlation ID 0x02=Incorrect Return Error 0x03=Unrecognized Error 0x04=Unexpected Error 0x05=Incorrect Parameter **Transaction Portion** 0x01=Unrecognized Package Type 0x02=Incorrect Transaction Portion 0x03=Badly Structured Transaction Portion 0x04=Unrecognized Transaction ID 0x05=Permission to Release 0x06=Resource Unavailable

Protocol Version

The values for the Protocol Version parameter are automatically stored, consistent with ITU TCAP specifications. They cannot be modified by the host if you use the ITU White Book.

Quality of Service (QOS)

Byte	Value(s)
Data[4] Return Option	0x00=Discard on Error 0x01=Return on Error
Data[5] Sequence Control Parameter	0x00=Sequence not Guaranteed For other than zero, this value is used for sequence control.**
Data[6] Message Priority*	(ANSI Only)

*Refer to ANSI MTP Specification T1-111.5 for the information on selecting message priority.

**For the outgoing messages, the sequence is guaranteed for the messages with the same sequence control value. They will reach the destination in the same order as they were sent out. For the incoming messages, this value simply means the sequence is guaranteed (SCCP class 1 services are used in this case).

Remote SCCP Status

Byte	Value(s)
Data[4] Status	0x01=Remote SCCP Available 0x02=Remote SCCP Unavailable (Reason Unknown) 0x03=Remote SCCP Unequipped 0x04=Remote SCCP Inaccessible

Return Reason (ITU)

Byte	Value(s)
Data[4] Cause	0x00 No Translation for an Address of Such Nature 0x01 No Translation for this Specific Address 0x02=Subsystem Congestion 0x03=Subsystem Failure 0x04=Unequipped User 0x05=MTP Failure 0x06=Network Congestion 0x07=Unqualified 0x08=Error in Message Transport * 0x09=Error in Local Processing * 0x0A=Destination Cannot Perform Reassembly * 0x0B=SCCP Failure *Only applicable to XUDTS message.

Return Reason (ANSI)

Byte	Value(s)
Data[4] Cause	0x00 No Translation for an Address of Such Nature 0x01 No Translation for this Specific Address 0x02=Subsystem Congestion 0x03=Subsystem Failure 0x04=Unequipped User 0x05=MTP Failure 0x06=Network Congestion 0x07=Unqualified 0x08=Error in Message Transport * 0x09=Error in Local Processing * 0x0A=Destination Cannot Perform Reassembly * 0x0B=Not Used 0x0C=SCCP Hop Counter Violation * 0x0D – F8=Not Used 0xF9=Invalid ISNI Routing Request * 0xFA=Unauthorized Message 0xFB=Message Incompatibility 0xFC=Cannot Perform ISNI Constrained Routing * 0xFD=Redundant ISNI Constrained Routing Information * 0xFE=Unable to Perform ISNI Identification * 0xFF=Not Used *=Only applicable to XUDT and XUDTS messages.

SCCP Signaling Point Status

Byte	Value(s)
Data[4] Status	0x01=SCCP DPC Prohibited 0x02=SCCP DPC Congested 0x03=SCCP DPC Allowed

Subsystem Status

Byte	Value(s)
Data[4] Status	0x00=Prohibit 0x01=Allow

Termination Option

Byte	Value(s)
Data[4] Option	0x01=Pre-arranged End 0x02=Basic End

Unidialog_AS_ID

The values for the Unidialog_AS_ID parameter are automatically stored, consistent with ITU TCAP specifications. They should not be modified by the host if you use the ITU White Book.

User-Defined Parameters

- Application Context Name
- MTP DPC
- Correlation ID
- Invoke ID
- Linked ID
- Parameters
- Parameter Sequence
- Parameter Set
- Signaling Point Code (SPC)
- Subsystem Number
- Timeout
- User Data
- User Information

SCCP Segmentation

Overview

SCCP segmentation provides the capability of breaking down larger packets of data into smaller ones in order to achieve compatibility with any network protocol that requires the smaller packet size. This is necessary whenever large blocks of data need to be transmitted across a network Segmenting the data helps offset problems with both time delays and error correction that could lead to traffic congestion. Segmentation in this way conserves critical network resources.

Specifically, the process of SCCP Segmentation gives an SCCP-user the ability to configure and send user data without having to perform any segmentation at the user level. SCCP handles all segmentation requirements.

Size of Packetized Data

The size of data sent to the host is currently limited by the ICB length. The ICB length is a maximum of 255 bytes in API messages. Connectionless transfer of data using SCCP requires the use of unitdata and the extended unitdata structures. These message structures provide all the information necessary for data to be transferred to a remote entity and to be processed by that remote entity. The segmentation parameter is included with these messages in an SCCP connectionless service:

- Extended UnitData (XUDT, (maximum of 16)
- Extended UnitData Service (XUDTS)

For an SCCP user resident in the host, the size of the total data sent between the SS7 PQ card and the host will be limited by the maximum length of the ICB, which is 255 bytes, until the system level segmentation has been implemented.

No segmentation occurs at the user level; therefore, an SCCP user is able to send up to 2048 bytes of data for ITU protocol and up to 3092 bytes of user data for the ANSI protocol.

Related Specifications

- ITU-T Q.711 - Q.714: Signaling Connection Control Part, White Book
- ANSI T1.112 - 1996

Managing User Data in SCCP

Data sent by an SCCP user can be limited by the constraints inherent in the MTP layer, that is, the maximum length of the Message Signal Unit (MSU). Through segmentation, the user is able to transmit through

MTP efficiently because the data is segmented into manageable units called XUDT messages. When SCCP receives the XUDT messages, it reassembles all data into an N_UNITDATA primitive and forwards that to the SCCP user/receiver.

SCLC, SCRC and SSI Modules

It is the SCCP SCLC module that segments the N_UNITDATA primitive into XUDT messages and which also performs the reassembly of data. The SCRC module performs the routing of these connectionless messages. The SCCP SUSI module receives the reassembled message and sends a larger UNITDATA PPL Event Indication to the SCCP user (whether on the CSP or resident on the host).

Error Conditions

If the SCCP cannot process the XUDT message because an error condition has occurred on the network, and the return option is set, then it gets an XUDTS message and forwards notification of the error to the SCCP user at the sending end using an N_ NOTICE primitive. (See the ITU-T Q.711 - Q.714: Signaling Connection Control Part, White Book and ANSI T1.112 - 1996).

Configuration

Configuration bytes for the TCAP module in the *PPL Configure* message must be set for each component ID. With SCCP Segmentation implemented, these config bytes allow the TCAP layer to send larger blocks of data to the SCCP layer if desired. SCCP can send or receive messages directly with the host or through the TCAP layer. The configuration bytes are discussed further in the SS7 PPL Information chapter, and include the following:

- SCCP Component SCLC--Bytes 22 and 26 (a 4-byte value)
- SCCP Component SUSI--Byte 20
- TCAP Component CCO--Bytes 1 and 2
- TCAP Component TUSI--Bytes 4 and 5

Limitations

SS7 Segmentation allows a maximum number of 300 reassembly processes simultaneously.

Redundancy

SCCP software is redundant.

TCAP Overload Logic

Purpose

The Transactional Capabilities Application Part (TCAP) Overload Logic feature provides overload condition detection of SS7 card resources in the CSP. This feature will prevent the SS7 card from failing under overload conditions by reducing TCAP traffic on the card.

Resources Monitored for Overload Conditions

To provide TCAP overload detection according to traffic busy levels, the following resources will be monitored for overload conditions:

- CPU usage
- Memory usage
- Message Control Blocks (MCBs) usage

TCAP Overload Condition Detection

The TCAP Overload Logic feature provides for four conditions to be detected:

- Approaching Busy
- Real Busy
- Real Busy Clear
- Approaching Busy Clear

Each condition listed above will be detected for the CPU usage, memory usage and MCB usage resources. As long as one of the resources is going into a BUSY state, for example APPROACHING BUSY, the system will be considered to be in that corresponding state. When a resource is going into a CLEAR state, for example APPROACHING BUSY CLEAR, the system will be considered to be in that corresponding state.

Call Processing Filter Facility

In order for the TCAP overload logic to maintain a stable state for a period of time and avoid frequent changes among the four conditions to be detected, a filter facility is provided.

The filter facility uses five APPROACHING BUSY levels to indicate the busy conditions. The higher level, the busier the condition. Busy level 0 indicates a not busy condition and busy level 4 indicates the busiest condition.

Every time an APPROCHING BUSY condition is detected, the busy level is increased by one. TCAP will reject any new TCAP Transaction

Requests by sending an ABORT message to the network to abort a portion of new TCAP transactions (Begin [ITU/ETSI] or Query [ANSI]) and send a NACK to the host for new transaction requests (TC-Begin/TC-Query) according to the busy level. Based on the increase of the busy level, more traffic will be discarded and NACKED. For example, when the busy level is 1, one of every four calls will be discarded and NACKED, and if the busy level increases to 2, then two of every four calls will be discarded and NACKED. Conversely, a decrease in the busy level will result in more calls being processed by TCAP. When the busy level is zero, all of the TCAP calls will be processed.

TCAP provides a timer interrupt to detect and find the BUSY or CLEAR conditions by monitoring CPU usage, memory usage and MCB usage.

Busy and Clear Conditions

The following describes the process of handling BUSY or CLEAR conditions.

Approaching Busy

When an APPROACHING BUSY condition is detected for incoming new transaction requests, TCAP will send an ABORT message to the remote SS7 node that is running the TCAP on the network. TCAP will abort a portion of the new TCAP transactions (Begin/Query).

For the outgoing new transaction requests, TCAP will send a NACK to the host, also based on the APPROACHING BUSY level and the filter facility busy level. TCAP will also send an ALARM message to the host.

Approaching Busy Clear

When an APPROACHING BUSY CLEAR condition is detected, it is sent to SCAP/TCAP. At this point TCAP will resume call processing for the incoming direction, and ACKnowledge the host's request for the outgoing direction based on the filter facility busy level. TCAP will also send an ALARM CLEAR message to the host to indicate a busy clear condition.

Real Busy

When TCAP detects a REAL BUSY condition, TCAP will instruct the Message Transfer Port level 3 (MTP3) to discard incoming calls and and free the memory. TCAP will also send a NACK message and an

ALARM message to the host. TCAP will resume processing TCAP traffic when the resource busy condition goes away.

Real Busy Clear

When TCAP detects a REAL BUSY CLEAR condition, it will send an ALARM message to the host, and resume the TCAP call processing. TCAP will still send a NACK message to the host for new transaction requests, until the APPROCHING BUSY CLEAR condition is received.

Busy Condition Alarm Levels to Host

There are two different levels of busy alarms that will be sent to the host when a BUSY condition is present or clears.

- Card Level Alarm

The card level alarm is sent to the host by TCAP and indicates the SS7 card is busy. The following alarm types are used:

ALARMbrdCARD_APPROACHING_BUSY and ALARMbrdCARD_REAL_BUSY,

- Stack and Application Level Alarm

The stack and application level alarm is sent to the host by the SCAP and indicates the SCAP/TCAP stack is busy. The following alarm type is used:

ALARMgenSS7_SIGNALING_STACK_BUSY.

TCAP Active and Standby SS7 Card Redundancy - CSP

The TCAP call processing logic on redundant SS7 cards provides that when a busy condition exists on the active SS7 card, there also exists a busy condition on the standby SS7 card. TCAP calls are first processed in the TCAP of the active SS7 card, then the message will be forwarded to the TCAP of the standby SS7 card for processing. While the TCAP in the active card cuts down the traffic, the TCAP in the standby will also cut down the traffic too.

Configuring TCAP Overload Logic

Purpose

This section describes how to change the threshold for TCAP overload logic configuration on your SS7 card in the CSP whether configured for a variant of an ANSI or ITU protocol.

Before you begin

You must have an SS7 card configured with TCAP and TCAP must have a local subsystem. You can change either the threshold for approaching busy or real busy. You must ensure that the real busy threshold is set at higher rate than the approaching busy threshold.

Configuring TCAP Overload Logic

Use the steps below to change the threshold for TCAP Overload Logic.

1 Configure the threshold using the *System Configuration (0x00AF)* message.

2 Depending on the threshold being changed, set the Configuration Type to either:

- 0x0B
- 0x0C

3 Use the slot AIB for the slot number of the SS7 card.

4 Set the data. For example, the data relevant to the real busy threshold for CPU Level 1 would be:

02 CPU level 1

01 Enable

02 Change thresholds

62 CPU usage percentage expressed in hexadecimal.

Important! Dialogic recommends you not change the thresholds for Memory and MCB.

5 See the next example.

END OF STEPS

Example System Configuration message

The following shows the System Configuration (0x00AF) message being used to change the CPU (Level 1) real busy threshold for slot 0 to 98 per cent:

00 0f 00 af 00 00 ff 0c 00 01 01 01 00 02 01 02 62

7 SS7 PPL Information

Purpose This chapter outlines the PPL components in the SS7 stack of the CSP system software that are available for customizing within the following SS7 modules:

- L3P
- ISUP
- TUP
- L3P SSUTR2
- BT IUP
- MTP3
- MTP2 TXC (0x26)
- SCCP/TCAP

> **Important!** Unless indicated otherwise, it is assumed that all configuration bytes are part of Block 1

With each module, specific information can be found within PPL component IDs according to ITU and/or ANSI standards: Configuration Bytes, PPL Events and PPL Timers. For some components, only PPL Timers can be changed.

For an explanation of PPL formats and configuration, see the *Developer's Guide: Overview.*

For a list of the PPL Components and addressing information, see PPL Component IDs and PPL Component Addressing in the *API Reference*.

L3P CIC (0x000F)

Purpose

This section includes ITU and ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component L3P CIC.

L3P CIC Configuration Bytes (ITU)

The table below shows the ITU PPL Configuration Byte values for the L3P CIC component. See *Important ISUP PPL Information (4-112)* for an explanation of config byte mapping for L3P CIC.

You can modify the parameters for Config Bytes 1-7, but you cannot move them to other byte locations. Other functions in the system access these parameters.

Block 1

Byte	Description	Value
0x01	Index to CGB/CGU parameters	0x0F
0x02	Index to Hardware CGB/CGU parameters	0x19
0x03	Index to GRS parameters	0x22
0x04	Called Party Nature of Address Indicator	0x03 (Subscriber Number)
0x05	Called Party Numbering Plan	0x10 (Private Numbering Plan)
0x06	Calling Party Nature of Address Indicator	0x00 (Subscriber Number)
0x07	Calling Party: Numbering Plan Address Presentation Restriction Indicator Screening Indicator (SI)	0x15 (Private Numbering Plan,PresentationRestricted, User Provided/Screening Passed)
0x08	Request For Service With Data format	0x00=Raw ISUP (default) 0x01=BCD digits
0x09	Incoming Call Processing Option	0x00=ACM/ANM (default) 0x01=CON

Byte	Description	Value
0x0A	CIC Out of Service/In Service Transition Flag	0x00=OOS Send BLO, INS Send UBL (default) 0x01=OOS No Action, INS Send RSC 0x02=No Action (default for Exchange Type A)
0x0B	State Transitions Operation Flag	0x00=Group Operations (default) 0x01=Individual Operations
0x0C	Outgoing Congestion Control	0x00=Disable 0x02=Enable (default)
Miscellaneous Configuration		
0x0D	Misc. Config Byte: Send UPT automatically upon UPU from MTP	00=Do not send UPT 01=Send UPT automatically
0x0E	Host Indication Flag	0x00=Do Not Send PPL Event to Host 0x01=Send PPL Event to Host (default)
Circuit Group Blocking/Circuit Group Unblocking (CGB/CGU) (Maintenance)		
0x0F	Information Length	0x07
0x10	Message ID	0x00 (CGB/CGU)
0x11	Number of Parameters	0x02
0x12	Parameter 1 ID	0x15 (C.G.S.M.T. Indicator)
0x13	Parameter 1 Data Length	0x01
0x14	Parameter 1 Data [0]	0x00 (Block with Immediate Release)
0x15	Parameter 2 ID	0x16 (Range & Status)
0x16	Parameter 2 Data Length	0x00
0x17	Priority Call Handling when Outgoing Congestion is true	0 - Do not allow priority calls 1 - Allow priority calls
0x18	Unused	0x00

Byte	Description	Value
Circuit Group Blocking/Circuit Group Unblocking (CGB/CGU) (Hardware)		
0x19	Information Length	0x07
0x1A	Message ID	0x00 (CGB/CGU)
0x1B	Number of Parameters	0x02
0x1C	Parameter 1 ID	0x15 (C.G.S.M.T. Indicator)
0x1D	Parameter 1 Data Length	0x01
0x1E	Hardware Orientation	0x01
0x1F	Parameter 1 ID	0x16 (Range & Status)
0x20	Parameter 2 Data Length	0x00
0x21	Unused	0x00
Circuit Group Reset (GRS)		
0x22	Information Length	0x04
0x23	Message ID	0x00 = Behavior determined by this configuration byte and is pre-95 ANSI. 0x01 = Behavior is determined by the Message Configuration Template by default.
0x24	Number of Parameters	0x01
0x25	Parameter 1 ID	0x16 (Range & Status)
0x26	Parameter 1 Data Length	0x00
0x29	PPL Event Indications sent to the host for ACM/ANM.	0x00* = Send 0x01 = Do not send

Byte	Description	Value
0x2B	Send RFS to host before incoming Continuity result is received and connect loopback without OOS. In this case, the host should not try to send any other call processing API on the circuit where incoming Continuity Check is being performed, until continuity check results are received in the form of PPL Event Indication.	0x00 = Send DS0 status change of Channel Out of Service (Loopback) to the host. 0x01 = Do not send DS0 status change of Channel Out of Service (Loopback) to the host
0x2C	Determines whether the outgoing channel receives the *Channel Release with Data* (0x0069) message when the outgoing channel is released by the other end before the Address Complete Message (ACM).	0x00=Receive *Channel Release with Data* (0x0069) (default) 0x01=Does not receive *Channel Release with Data* (0x0069)
0x2D	Determines whether Outseize ACK message is sent before or after the Address Complete Message (ACM).	0x00=Sent after ACM 0x01=Sent before ACM
0x2E	Determines whether SS7 Address Information is sent to Layer 4.	0x00=Not sent 0x01=Sent
0x2F	Determines whether to re-transmit the Blocking (BLO) message when the RSC is sent to a LO BLO circuit.	0x00=Re-transmits 0x01=Does not re-transmit
0x30	Determines whether the SAM in the RFS feature is enabled. Note: If you change this byte to 0x01, you must also set component 0x0012, Block 2, byte 0x0A to value 0x01	0x00=Disabled (default) 0x01=Enabled
0x31	Maximum digits to be collected, excluding the “End of Dialing/digits” signal [i.e., Signal Terminator (ST)]	Recommended range: 0x00-0x28 (The Digit_Buffer has 80 digit maximum.) 0x0A (Default)
0x32	ST digit value	0x0F (default)

Byte	Description	Value
0x33-0x36	Unused	
Address Complete Message (ACM)		
0x3C	Information Length	0x06
0x3D	Message ID	0x06 (ACM)
0x3E	Number of Parameters	0x01
0x3F	Parameter 1 ID	0x11 (Backward Call Indicators)
0x40	Parameter 1 Data Length	0x02
0x41	Parameter 1 Data[0]	0x00 (Charge=No Indicator Called Status=No Indicator Called Count=No Indicator End-End=No Method)
0x42	Parameter 1 Data[1]	0x04 (Interworking=None Segmentation=None ISUP=All the Way Hold=Not Required Term Access=Non-ISDN Echo=Not Included SCCP Method=No Indicator)
0x43–0x45	Unused	0x00
Answer Message (ANM)		
0x46	Information Length	0x02
0x47	Message ID	0x09 (ANM)
0x48	Number of Parameters	0x00
0x49–0x4A	Unused	0x00

Byte	Description	Value
Call Progress (CPG)		
0x4B	Information Length	0x05
0x4C	Message ID	0x2C (CPG)
0x4D	Number of Parameters	0x01
0x4E	Parameter 1 ID	0x24 (Event Information)
0x4F	Parameter 1 Data Length	0x01
0x50	Parameter 1 Data[0]	0x02 (Progress)
0x51–0x54	Unused	0x00
Connect (CON)		
0x55	Information Length	0x06
0x56	Message ID	0x07 (CON)
0x57	Number of Parameters	0x01
0x58	Parameter 1 ID	0x11 (Backward Call Indicator)
0x59	Parameter 1 Data Length	0x02
0x5A	Parameter 1 Data[0]	0x00 (Charge=No Indicator Called Status=No Indicator Called Count=No Indicator End-End=No Method)

Byte	Description	Value
0x5B	Parameter 1 Data[1]	0x14 (Interworking=None Segmentation=None ISUP=All the Way Hold=Not Required Term Access=ISDN Echo=Not Included SCCP Method=No Indicator)
0x5C–0x5E	Unused	0x00
Release (REL)		
0x5F	Information Length	0x06
0x60	Message ID	0x0C (REL)
0x61	Number of Parameters	0x01
0x62	Parameter 1 Data Type	0x12 (Cause Indicators)
0x63	Parameter 1 Data Length	0x02 (Cause Indicators)
0x64	Parameter 1 Data[0]	0x80 (Location=User)
0x65	Parameter 1 Data[1]	0X9F (Cause 31=Normal, Unspecified)
0x66 – 0x6D	Unused	0X00
Blocking (BLO)		
0x6E	Information Length	0x02
0x6F	Message ID	0x13 (BLO)
0x70	Number of Parameters	0x00
0x71– 0x72	Unused	0x00
Unblocking (UBL)		
0x73	Information Length	0x02
0x74	Message ID	0x14 (UBL)
0x75	Number of Parameters	0x00

Byte	Description	Value
0x76– 0x81	Unused	0x00
Initial Address Message (IAM)		
0x82	Information Length	0x0F
0x83	Message ID	0x01 (IAM)
0x84	Number of Parameters	0x04
0x85	Parameter 1 ID	0x06 (Nature of Connection)
0x86	Parameter 1 Data Length	0x01
0x87	Parameter 1 Data[0]	0x04 - enables continuity on the outgoing side 0x08 - enables continuity of the previous circuit Refer to Spec. Q.763 parameter 0x06 Nature of Connection, Paragraph 3.35 for full explanation of all bits.
0x88	Parameter 2 ID	0x07 (Forward Call Indicators)
0x89	Parameter 2 Data Length	0x02
0x8A	Parameter 2 Data[0]	0x22 (Call=National End-End=Pass along method available Interworking=None Segmentation=None ISUP=All the Way Preference=ISUP)

Byte	Description	Value
0x8B	Parameter 2 Data[1]	0x00 (Original Access=Non-ISDN SCCP Method=No Indicator Called Number=No Translate QOR=No Attempt)
0x8C	Parameter 3 ID	0x09 (Calling Party Category)
0x8D	Parameter 3 Data Length	0x01
0x8E	Parameter 3 Data[0]	0x0A
0x8F	Parameter 4 ID	0x02 (Transmission Medium Requirements)
0x90	Parameter 4 Data Length	0x01=Speech
0x91	Parameter 4 Data[0]	0x00=Speech
0x92 to 0xA6	Unused or User-Defined	
0xA7	If PPL Event Request of ANM is sent to L3P to automatically put, L4 in the answer state.	0 - Do not put L4 in answered state automatically (default) 1 - Put L4 in answered state automatically
0xA8-0xC8	Unused or User-Defined	

Block 2

The table below shows the config bytes for Block 2.

Byte	Description	Value
Call Modification Request (CMR)		
0x00-0x45	Unsed or User-Defined	
0x46	Information Length	0x05
0x47	Message ID	0x1C (CMR)
0x48	Parameter Counts	0x01
0x49	Parameter 1 ID	0x17 (Call Modification Indicators)
0x4A	Parameter 1 Data Length	0x01

Byte	Description	Value
0x4B	Parameter 1 Data[0]	0x01
Call Modification Complete (CMC)		
0x50	Information Length	0x05
0x51	Message ID	0x1D (CMC)
0x52	Parameter Count	0x01
0x53	Parameter 1 ID	0x17 (Call Modification Indicators)
0x54	Parameter Data Length	0x01
0x55	Parameter 1 Data[0]	0x01
Call Modification Reject (CMRJ)		
0x5A	Information Length	0x05
0x5B	Message ID	0x1E (CMRJ)
0x5C	Parameter Count	0x01
0x5D	Parameter 1 ID	0x17 (Call Modification Indicators)
0x5E	Parameter 1 Data Length	0x01
0x5F	Parameter 1 Data[0]	0x01
User Part Test (TPT)		
0x64	Information Length	0x06
0x65	Message ID	0x34 (UPT)
0x66	Number of Parameters	0x01
0x67	Parameter 1 ID	0x39 (Parameter Compatibility Information)
0x68	Parameter 1 Length	0x02
0x69	Parameter 1 Data[0]	0x39
0x6A	Parameter 1 Data[1]	0x00 (Not Used)
Calling Party Clear (CCL) (China ISUP)		
0x6E	Information Length	0x06
0x6F	Message ID	0xFC (CCL)
0x70	Number of Parameters	0x01
0x71	Parameter 1 ID	0x38 (Message Compatibility Information)
0x72	Parameter 1 Length	0x02

Byte	Description	Value
0x73	Parameter 1 Data[0]	0xFC (Transit Interp, Do Not Release Call, Notify, Discard Message)
0x74	Parameter 1 Data[1]	0x00
Operator (OPR) (China ISUP)		
0x78	Information Length	0x06
0x79	Message ID	0xFE (OPR)
0x7A	Number of Parameters	0x01
0x7B	Parameter 1 ID (Message Compatibility Information)	0x38 (Message Compatibility Information)
0x7C	Parameter Length	0x02
0x7D	Parameter Data[0]	0xFE (Transit Interp, Do Not Release Call, Notify, Discard Message)
0x7E	Parameter Data[1]	0x00 (Not Used)
Forward Transfer (FOT)		
0x82	Information Length	0x02
0x83	Message ID	0x08 (FOT)
0x84	Number of Parameters	0x00
Delayed Release (DRS)		
0x8C	Information Length	0x02
0x8D	Message ID	0x27 (DRS)
0x8E	Number of Parameters	0x00
Meter Pulse Message (MPM) (China ISUP)		
0x96	Information Length	0x06
0x97	Message ID	0xFD (MPM)
0x98	Number of Parameters	0x01

Byte	Description	Value
0x99	Parameter 1 ID	0xFE (Charge Information)
0x9A	Parameter 1 Length	0x02
0x9B	Parameter 1 Data[0]	0x01
0x9C	Parameter 1 Data[1]	0x00

Block 3

The table below shows the config bytes for Block 3.

Byte	Description	Value
Circuit Query Message (CQM)		
0x01	Information Length	0x04
0x02	Message ID	0x00 (CQM)
0x03	Number Of Parameters	0x01
0x04	Parameter 1 ID	0x16 (Range and Status)
0x05	Parameter 1 Data Length	0x00
Facility Request (FAR)		
0x0B	Information Length	0x05
0x0C	Message ID	0xIF(FAR)
0x0D	Number Of Parameters	0x01
0x0E	Parameter 1 ID	0x18 (Facility Indicator)
0x0F	Parameter 1 Data Length	0x01
0x10	Parameter 1 Data[0]	0x02 (User-to-User Services)
Facility Accept (FAA)		
0x11	Information Length	0x05
0x12	Message ID	0x20 (FAA)
0x13	Number Of Parameters	0x01
0x14	Parameter 1 ID	0x18 (Facility Indicator)
0x15	Parameter 1 Data Length	0x01
0x16	Parameter 1 Data[0]	0x02 (User-to-User Service)
Facility Reject (FRJ)		
0x17	Information Length	0x09

Byte	Description	Value
0x18	Message ID	0x21 (FRJ)
0x19	Number Of Parameters	0x02
0x1A	Parameter 1 ID	0x18 (Facility Indicator)
0x1B	Parameter 1 Data Length	0x01
0x1C	Parameter 1 Data[0]	0x02 (User-to-User Service)
0x1D	Parameter 2 ID	0x12 (Cause Indicators)
0x1E	Parameter 2 Data Length	0x02
0x1F	Parameter 2 Data [0]	0x80 (Location=User)
0x20	Parameter 2 Data [1]	0xC5 (Cause 69=Requested Facility Not Implemented)

Byte	Description	Value
Facility (FAC)		
0x21	Information Length	0x02
0x22	Message ID	0x33 (FAC)
0x23	Number Of Parameters	0x00
User (USR)		
0x24	Information Length	0x06
0x25	Message ID	0x2D (USR)
0x26	Number Of Parameters	0x01
0x27	Parameter 1 ID	0x20 (User-to-User Information)
0x28	Parameter 1 Data Length	0x02
0x29	Parameter 1 Data[0]	0xFF (All 1s)
0x2A	Parameter 1 Data[1]	0xFF (All 1s)
0x2B	Unused	
Network Resource Management (NRM)		
0x2C	Information Length	0x05
0x2D	Message ID	0x32 (NRM)
0x2E	Number Of Parameters	0x01
0x2F	Parameter 1 ID	0x37 (Echo Control Information)
0x30	Parameter 1 Data Length	0x01
0x31	Parameter 1 Data	0x55 (Out Device Not Included/Not Available, In Device Not Included/Not Available, Out Device Activation Request, In Device Activation Request)
Pass Along Message (PAM)		
0x32	Information Length	0x01
0x33	Message ID	0x28 (PAM)
Identification Request (IDR)		

Byte	Description	Value
0x35	Information Length	0x02
0x36	Message ID	0x36 (IDR)
0x37	Number Of Parameters	0x00
Identification Response (IRS)		
0x38	Information Length	0x02
0x39	Message ID	0x37 (IRS)
0x3A	Number Of Parameters	0x00
Information Request (INR)		
0x3B	Information Length	0x06
0x3C	Message ID	0x03 (INR)
0x3D	Number Of Parameters	0x01
0x3E	Parameter 1 ID	0x0E (Information Request Indicator)
0x3F	Parameter 1 Data Length	0x02
0x40	Parameter 1 Data[0]	0x01 (Calling Party Address Requested, Holding Not Requested, Calling Party's Category Not Requested, Charge Information Not Requested)
0x41	Parameter 1 Data[1]	0x00 (Not Used)
Information (INF)		
0x42	Information Length	0x06
0x43	Message ID	0x04 (INF)
0x44	Number Of Parameters	0x01
0x45	Parameter 1 ID	0x0f (Information Indicator)
0x46	Parameter 1 Data Length	0x02

Byte	Description	Value
0x47	Parameter 1 Data[0]	0x00 (Calling Party Address Not Included, Hold Not Provided, Calling Party's Category Not Included, Charge Information Not Included, Solicited)
0x48	Parameter 1 Data[1]	0x00 (Not used)
Charge (CRG)		
0x49	Information Length	0x02
0x4A	Message ID	0x31 (CRG)
0x4B	Number Of Parameters	0x00
Subsequent Address Message (SAM)		
0x4C	Information Length	0x06
0x4D	Message ID	0x02 (SAM)
0x4E	Number Of Parameters	0x01
0x4F	Parameter 1 ID	0x05 (Subsequent Number)
0x50	Parameter 1 Data Length	0x02
0x51	Parameter 1 Data[0]	0x80 (Odd Number of Address Signals)
0x52	Parameter 2 Data[1]	0x0F (ST-End of Digits)
Suspend (SUS)		
0x55	Information Length	0x05
0x56	Message ID	0x0D (SUS)
0x57	Number Of Parameters	0x01
0x58	Parameter 1 ID	0x22 (Suspend/Resume Indicators)
0x59	Parameter 1 Data Length	0x01
0x5A	Parameter 1 Data[0]	0x01 (Network Initiated)
Resume (RES)		
0x5C	Information Length	0x05

Byte	Description	Value
0x5D	Message ID	0x0E (RES)
0x5E	Number Of Parameters	0x01
0x5F	Parameter1 ID	0x22 (Suspend/Resume Indicators)
0x60	Parameter 1 Data Length	0x01
0x61	Parameter 1 Data[1]	0x01 (Network Initiated)
0x64-0xC8	See Table below.	

L3P CIC Config Bytes with Offset Values (ITU)

Config Byte Block 3 is used for the config byte range of 400-600, where the offset table resides.

:

Index	Message	Config Byte	Value
1	Misc. Info	0x65	1 (Offset Block No.)
		0x66	0x01 (Offset Range: 0x01-0x0E)
2	ACM	0x67	1 (Offset Block No.)
		0x68	0x3C (Offset Range: 0x3C-0x42)
3	ANM	0x69	1 (Offset Block No.)
		0x6A	0x46 (Offset Range: 0x46-0x48)
4	CPG	0x6B	1 (Offset Block No.)
		0x6C	0x4B (Offset Range: 0x4B-0x50)
5	CON	0x6D	1 (Offset Block No.)
		0x6E	0x55 (Offset Range: 0x55-0x5B)
6	REL	0x6F	1 (Offset Block No.)

Index	Message	Config Byte	Value
		0x70	0x5F (Offset Range: 0x5F-0x65)
7	BLS	0x71	1 (Offset Block No.)
		0x72	0x6E (Offset Range: 0x6E-0x70)
8	UBL	0x73	1 (Offset Block No.)
		0x74	0x73 (Offset Range: 0x73-0x75)
9	IAM	0x75	1 (Offset Block No.)
		0x76	0x82 (Offset Range: 0x82-0x90)
10	CQM	0x77	3 (Offset Block No.)
		0x78	0x01 (Offset Range: 0x01-0x05)
11	FAR	0x79	3 (Offset Block No.)
		0x7A	0x0B (Offset Range: 0X0B-0X10)
12	FAA	0x7B	3 (Offset Block No.)
		0x7C	0x11 (Offset Range: 0x11-0X16)
13	FRJ	0x7D	3 (Offset Block No.)
		0x7E	0x17 (Offset Range: 0x17-0x20)
14	FAC	0x7F	3 (Offset Block No.)
		0x80	0x21 (Offset Range: 0x21-0x23)
15	USR	0x81	3 (Offset Block No.)

Index	Message	Config Byte	Value
		0x82	0x24 (Offset Range: 0x24--0x2A)
16	NRM	0x83	3 (Offset Block No.)
		0x84	0x2C (Offset Range: 0x2C-0x31)
17	PAM	0x85	3 (Offset Block No.)
		0x86	0x32 (Offset Range: 0x32-0x34)
18	IDR	0x87	3 (Offset Block No.)
		0x88	0x35 (Offset Range: 0x35-0x37)
19	IRS	0x89	3 (Offset Block No.)
		0x8A	0x38 (Offset Range: 0x38-0x3A)
20	INR	0x8B	3 (Offset Block No.)
		0x8C	0x3B (Offset Range: 0x3B-0x41)
21	INF	0x8D	3 (Offset Block No.)
		0x8E	0x42 (Offset Range: 0x42-0x48)
22	CRG	0x8F	3 (Offset Block No.)
		0x90	0x49 (Offset Range: 0x49-0x4B)
23	SAM	0x91	3 (Offset Block No.)

Index	Message	Config Byte	Value
		0x92	0x4C (Offset Range: 0x4C-0x52)
24	SUS	0x93	3 (Offset Block No.)
		0x94	0x55 (Offset Range: 0x55-0x5A)
25	RES	0x95	3 (Offset Block No.)
		0x96	0x5C (Offset Range: 0x5C-0x61)
26	UPT	0x97	3 (Offset Block No.)
		0x98	0x02 (Offset Range: 0x64-0x6A)
27	CCL	0x99	3 (Offset Block No.)
		0x9A	0x02 (Offset Range: 0x6E-0x74)
28	OPR	0x9B	3 (Offset Block No.)
		0x9C	0x02 (Offset Range: 0x78-0x7E)
29	FOT	0x9D	3 (Offset Block No.)
		0x9E	0x02 (Offset Range: 0x82-0x84)
30	DRS	0x9F	3 (Offset Block No.)
		0xA0	0x02 (Offset Range: 0x8C-0x8E)
31	MPM	0xA1	3 (Offset Block No.)
		0xA2	0x02 (Offset Range: 0x96-0x9C)
32	CMR	0xA3	3 (Offset Block No.)

Index	Message	Config Byte	Value
		0xA4	0x02 (Offset Range: 0x46-0x4B)
33	CMC	0xA5	3 (Offset Block No.)
		0xA6	0x02 (Offset Range: 0x50-0x55)
34	CMRJ	0xA7	3 (Offset Block No.)
		0xA8	0x02 (Offset Range: 0x5A-0x5F)
35	COT	0xA9	3 (Offset Block No.)
		0xAA	0x02 (Offset Range: 0x3C-0x3E)
36	LOP	0xAB	3 (Offset Block No.)
		0xAC	0x02 (Offset Range: 0xAA-0xAC)
37	SGM	0xAF	3 (Offset Block No.)
		0xB0	0x02 (Offset Range: 0xB4-0xB6)
	:	:	---
	:	---	---
	Undefined Message	0xC7	Offset Block N
	---	0xC8	---

L3P CIC PPL Event Indications (ITU)

The following table shows the PPL Event Indications supported by ITU ISUP L3P CIC.

Event ID	Event
0x01	Reset circuit success indication
0x08	Remote maintenance block or maintenance group block indication
0x09	Remote maintenance unblock or maintenance group unblock indication
0x0A	ACM received
0x0B	ANM received
0x0C	CPG received
0x0E	ISUP protocol violation
0x0F	Remote hardware group block indication
0x10	Remote hardware group unblock indication
0x11	CON received
0x12	Local maintenance block / group maintenance block success
0x13	Local maintenance unblock / group maintenance unblock success
0x15	Local circuit maintenance unblocking indication
0x16	UCIC received
0x19	OLM received
0x32	SGM message received

Event ID	Event
0x5A	SAM wait timer expired. The CSP receives an inbound call (IAM) and checks to see if the *Request for Service* message in the IAM is enabled (component 0x000F, byte 0x30 is set to 1). If so, the CSP checks if the number of maximum digits has been received (set in byte 0x31) and if it is not exceeded the CSP looks for the “f” digit in the called party number (or whatever is set in byte 0x32). If the maximum has not been received and the “f” was not included in the called party number the CSP starts timer 0x05 (component 0x000F) which is set to four seconds by default (unless changed in the configuration file). Upon expiration of Timer 0x05, this indication is sent to the host and then the RFS message is sent to the host.
0x63	DSP Resource Allocation Failure

L3P CIC PPL Event Requests (ITU)

The following table shows the PPL Event Requests supported by ITU ISUP L3P CIC.

Event ID	Event
0x01	Send ACM
0x02	Send CPG
0x03	Send ANM
0x04	Request local maintenance blocking (sends BLO)
0x05	Request local maintenance group blocking (sends CGB)
0x06	Request local maintenance unblocking (sends UBL)
0x07	Request local maintenance group unblocking (sends CGU)
0x08	Send GRS
0x09	Send RSC
0x0A	Send HW block
0x0B	Send HW unblock
0x0C	Indicate CPC to wait for incoming COT report when the continuity check is being performed on the previous circuit
0x0D	Send incoming continuity check report when the continuity check was performed on the previous circuit.
0x0E	Send unrecognized message to the network. It is used for message pass on during MCP handling. **NOTE:** ICB subtype 0x1F should be used for sending the unrecognized message. DPC/CIC in the ICB should be sent as “00”.
0x1E	Send CON
0x1F	Send CQM
0x20	Send FAR
0x21	Send FAA
0x22	Send FRJ
0x23	Send USR
0x24	Send FAC
0x25	Send PAM
0x26	Send NRM

Event ID	Event
0x27	Send CRG
0x28	Send IDR
0x29	Send IRS
0x2A	Send INR
0x2B	Send INF
0x2D	Send SUS
0x2E	Send RES
0x32	Send SAM
0x33	Reset timer T35 (for inbound SAM)
0x40	Manual Stop (for Continuity re-checking)
0x41	Send Continuity Check Request Message (CCR)
0x42	Send UPT
0x43	Send CCL
0x44	Send OPR
0x46	Send FOT
0x47	Send DRS
0x49	CMC Request with/without Parameters
0x4A	CMRJ request with/without Parameters.
0x4B	CMR Request with/without Parameters
0x4C	Request for COT
0x4D	Request for CCR
0x4E	Request for LPA
0x55	Loop Prevention
0x60	Send SGM

L3P CIC PPL Timers (ITU)

Timer Index	Definition	Default Value (10ms)
0x01	Wait for SS7 Network Response	36000
0x02	Wait for L4 Response	3000
0x03	Wait for L4 Maintenance Loopback ACK	3000
0x04	Wait for GRS. Used in OOS Protocol	500
0x05	InterDigit Timer	400

L3P CIC PPL Configuration Bytes (ANSI)

See *Important ISUP PPL Information (4-112)* for an explanation of config byte mapping for L3P CIC.

You can modify the parameters for Config Bytes 1-7, but you cannot move them to other byte locations. Other functions in the system access these parameters.

Block 1

Byte	Description	Value
0x01	Index to CGB/CGU parameters	0x0F
0x02	Reserved	0x00
0x03	Index to GRS parameters	0x1E
0x04	Called Party Nature of Address Indicator	0x01 (Subscriber Number)
0x05	Called Party Numbering Plan	0x50 (Private Numbering Plan)
0x06	Calling Party Nature of Address Indicator	0x01 (Subscriber Number)
0x07	Calling Party: Numbering Plan Address Presentation Restriction Indicator Screening Indicator (SI)	0x55 (Private Numbering Plan, Presentation Restricted, User Provided/Screening Passed)
0x08	Request For Service With Data format	0x00=Raw ISUP (default) 0x01=BCD digit

Byte	Description	Value
0x09	Unused	0x00
0x0A	CIC Out of Service/In Service Transition Flag	0x00=OOS Send BLO, INS Send UBL (default) 0x01=OOS No Action, INS Send RLC or GRA 0x02=No Action (default for Exchange Type A)
0x0B	State Transitions Operation Flag	0x00=Group Operations (default) 0x01=Individual Operations
0x0C	Outgoing Congestion Control	0x00=Disable 0x02=Enable (default)
Circuit Validation Test (CVT)		
0x0D	CVT Enable/Disable when UPU Received	0x00=Disable 0x01=Enable
0x0E	Host Indication Flag	0x00=Do Not Send PPL Event to Host 0x01=Send PPL Event to Host (default)
Circuit Group Blocking/Circuit Group Unblocking (CGB/CGU)		
0x0F	Information Length	0x07
0x00	Message ID	0x00 (CGB/CGU)
0x11	Number of Parameters	0x02
0x12	Parameter 1 ID	0x15 (C.G.S.M.T. Indicator)
0x13	Parameter 1 Data Length	0x00
0x14	Parameter 1 Data[0]	0x01 (Block with Immediate Release)
0x15	Parameter 2 ID	0x00 (Range & Status)
0x16	Parameter 2 Data Length	0x00
0x17	Priority Call Handling when Outgoing Congestion is true	0 - Do not allow priority calls 1 - Allow priority calls
0x18–0x1D	Unused	0x00
Circuit Group Reset (GRS)		

Byte	Description	Value
0x1E	Information Length	0x04
0x1F	Message ID	0x00 (GRS)
0x20	Number of Parameters	0x01
0x21	Parameter 1 ID	0x16 (Range & Status)
0x22	Parameter 1 Data Length	0x00
0x23	ANSI T1.113-1995 indicates the potential presence of an optional parameter called "circuit assignment map" in both Group Reset (GRS) and Group Reset Acknowledgement (GRA) messages. A pointer to optional parameters must be included in all ISUP messages that can contain an optional parameter.By default, the CSP adds this pointer. When connecting to network devices that are using the Pre-1995 format of the GRS / GRA messages, the presence of the pointer can cause problems. In order to not include the pointer on a per CIC basis, set this configuration byte to 0x01.	0x00 Post-95 (Default) 0x01 Pre-95
0x24–0x2A	Unused	0x00
0x2B	Send RFS to host before incoming continuity result is received and connect loopback without OOS. In this case, the host should not try to send any other call processing API on the circuit where an incoming continuity check is being performed, until the continuity check results are received in the form of PPL Event Indication.	0x00 = Send DS0 status change of Channel Out of Service (Loopback) to the host. 0x01 = Do not send DS0 status change of Channel Out of Service (Loopback) to the host
0x2C-0x31	Unused	0x00
Address Complete Message (ACM)		
0x32	Information Length	0x06
0x33	Message ID	0x06 (ACM)
0x34	Number of Parameters	0x01
0x35	Parameter 1 ID	0x11 (Backward Call Indicators)

Byte	Description	Value
0x36	Parameter 1 Data Length	0x02
0x37	Parameter 1 Data[0]	0x00 (Charge=No Indication Called Status=No Indication Called Cat=No Indication End-End=No Method)
0x38	Parameter 1 Data[1]	0x04 (Interworking=None Segmentation=None ISUP=All the way Hold=Not required Term Access=Non-ISDN Echo=Not included SCCP Method=No Indication)
0x39–0x40	Unused	0x00
Answer Message (ANM)		
0x41	Information Length	0x02
0x42	Message ID	0x09 (ANM)
0x43	Number of Parameters	0x00
0x44–0x4A	Unused	0x00
Call Progress (CPG)		
0x4B	Information Length	0x05
0x4C	Message ID	0x2C (CPG)
0x4D	Number of Parameters	0x01
0x4E	Parameter 1 ID	0x24 (Event Information)
0x4F	Parameter 1 Data Length	0x01
0x50	Parameter 1 Data[0]	0x02 (Progress)

Byte	Description	Value
0x51–0x59	Unused	0x00
Release (REL)		
0x5A	Information Length	0x06
0x5B	Message ID	0x0C (REL)
0x5C	Number of Parameters	0x01
0x5D	Parameter 1 ID	0x12 (Cause Indicators)
0x5E	Parameter 1 Data Length	0x02
0x5F	Parameter 1 Data[0]	0x80 (Location=User)
0x60	Parameter 1 Data[1]	0X9F (Cause 31=Normal, Unspecified)
0x61–0x63	Unused	0x00
Blocking (BLO)		
0x64	Information Length	0x02
0x65	Message ID	0x13 (BLO)
0x66	Number of Parameters	0x00
Unblocking (UBL)		
0x6E	Information Length	0x02
0x6F	Message ID	0x14 (UBL)
0x70	Number of Parameters	0x00
0x71–0x77	Unused	0x00
Initial Address Message (IAM)		
0x78	Information Length	0x21
0x79	Message ID	0x01 (IAM)
0x7A	Number of Parameters	0x04
0x7B	Parameter 1 ID	0x06 (Nature of Connection)
0x7C	Parameter 1 Data Length	0x01
0x7D	Parameter 1 Data[0]	0x00 (Satellite=None Continuity Check=Not Required Echo=Not Included)

Byte	Description	Value
0x7E	Parameter 2 ID	0x07 (Forward Call Indicators)
0x7F	Parameter 2 Data Length	0x02
0x80	Parameter 2 Data[0]	0x20 (Call=National End-End=No Method Interworking=None Segmentation=None ISUP=All the Way Preference=ISUP)
0x81	Parameter 2 Data[1]	0x00 (Original Access=Non-ISDN SCCP Method=No Indication Called Number=No Translate QOR=No Attempt)
0x82	Parameter 3 ID	0x09 (Calling Party Category)
0x83	Parameter 3 Data Length	0x01
0x84	Parameter 3 Data	0x00 (Unknown)
0x85	Parameter 4 ID	0x1D (User Service Information)
0x86	Parameter 4 Data Length	0x03
0x87	Parameter 4 Data[0]	0x80 (Coding Standard=ITU-T Information Transfer=Speech)
0x88	Parameter 4 Data[1]	0x90 (Transfer Mode=Circuit Rate=64kbls)
0x89	Parameter 4 Data[2]	0xA2 (User=Layer 1 u-law)

Byte	Description	Value
REL (Temp Failure) Parameters		
0x8A-0x95	Unused or User-defined	
0x96	Information Length	0x06
0x97	Message ID	0x0C (REL)
0x98	Number of Parameters	0x01
0x99	Parameter 1 ID	0x12 (Cause Indications)
0x9A	Parameter 1 Data Length	0x02
0x9B	Parameter 1 Data [0]	0x80
0x9C	Parameter 1 Data [1]	0xA9 (Temp Failure)
0x9D-0xA6	Unused or User-defined	
0xA7	If PPL Event Request of ANM is sent to L3P to automatically put, L4 in the answer state.	0 - Do not put L4 in answered state automatically (default) 1 - Put L4 in answered state automatically
0xA8-0xC8	Unused or User-defined	
Block 2		
0x01-0xC8	Unused or User-defined	
Block 3		
Information (INF)		
0x01	Information Length	0x06
0x02	Message ID	0x03 (INR)
0x03	Number of Parameters	0x01
0x04	Parameter 1 ID	0x0E (Parameter ID)
0x05	Parameter 1 Data Length	0x02
0x06	Parameter 1 Data [0]	0x01
0x07	Parameter 1 Data [1]	0x00
0x0A	Information Length	0x06
0x0B	Message ID	0x04 (INF)
0x0C	Number Of Parameters	0x01
0x0D	Parameter 1 ID	0x0f (Information Indicator)

Byte	Description	Value
0x0E	Parameter Data Length	0x02
0x0F	Parameter 1 Data[0]	0x00 Calling Party Address=Not Included Hold=None Calling party Count=Not Included Charge Information=Not Included Information=Solicited
0x10	Parameter 1 Data[1]	0x00 (MC Bus Group=Not Included)
Pass Along Message (PAM)		
0x14	Information Length	0x02
0x15	Message ID	0x28 (PAM)
0x16	Number of Parameters	0x00

Facility (FAC)		
0x19	Information Length	0x02
0x1A	Message ID	0x33 (FAC)
0x1B	Number Of Parameters	0x00
	CVT	
0x1E	Information Length	0x02
0x1F	Message ID	0xEC (CVT)
0x20	Number of Parameters	0x00
Circuit Query Message (CQM)		
0x28	Information Length	0x04
0x29	Message ID	0x00 (CQM)
0x2A	Number Of Parameters	0x01
0x2B	Parameter 1 ID	0x16 (Range and Status)
0x2C	Parameter 1 Data Length	0x00
Circuit Reservation Message (CRM)		

0x32	Information Length	0x05
0x33	Message ID	0xEA (CRM)
0x34	Number of Parameters	0x01
0x35	Parameter 1 ID	0x06 (Nature of Connection)
0x36	Parameter 1 Data Length	0x01
0x37	Parameter 1 Data [0]	0x00 (Satellite=None Continuity Check=Not Required Echo=Not Included)
Circuit Reservation Acknowledgment CRA)		
0x3C	Information Length	0x02
0x3D	Message ID	0xE9 (CRA)
0x3E	Number of Parameters	0x00
Exit Message (EXM)		
0x41	Information Length	0x02
0x42	Message ID	0xED (EXM)
0x43	Number of Parameters	0x00
Forward Transfer (FOT)		
0x46	Information Length	0x02
0x47	Message ID	0x08 (FOT)
0x48	Number of Parameters	0x00
Suspend (SUS)		
0x50	Information Length	0x05
0x51	Message ID	0x0D (SUS)
0x52	Number Of Parameters	0x01
0x53	Parameter 1 ID	0x22 (Suspend/Resume Indicator)
0x54	Parameter 1 Data Length	0x01
0x55	Parameter 1 Data[0]	0x01 (Network Initiated)
Resume (RES)		
0x5A	Information Length	0x05
0x5B	Message ID	0x0E (RES)

0x5C	Number Of Parameters	0x01
0x5D	Parameter 1 ID	0x22 (Suspend/Resume Indicator)
0x5E	Parameter 1 Data Length	0x01
0x5F	Parameter 1 Data[0]	0x01 (Network Initiated)

Block 3 With Index Table

Config Byte Block 3 is used for the config byte range of 400-600, where the offset table resides

Index	Message	Config Byte	Value
1	Misc. Info	0x65	1 (Offset Block No.)
		0x66	0x01 (Offset Range: 0x01-0x0E)
2	ACM	0x67	1 (Offset Block No.)
		0x68	0x32 (Offset Range: 0x32-0x38)
3	ANM	0x69	1 (Offset Block No.)
		0x6A	0x41 (Offset Range: 0x41-0x43)
4	CPG	0x6B	1 (Offset Block No.)
		0x6C	0x4B (Offset Range: 0x4B-0x50)
5	REL	0x6D	1 (Offset Block No.)
		0x6E	0x5A (Offset Range: 0x5A-0x60)
6	BLO	0x6F	1 (Offset Block No.)
		0x70	0x64F (Offset Range: 0x64-0x66)
7	UBL	0x71	1 (Offset Block No.)
		0x72	0x6E (Offset Range: 0x6E-0x70)
8	IAM	0x73	1 (Offset Block No.)
		0x74	0x78 (Offset Range: 0x78-0x88)
9	CQM	0x75	3 (Offset Block No.)

Index	Message	Config Byte	Value
		0x76	0x28 (Offset Range: 0x28-0x2C)
10	INR	0x77	3 (Offset Block No.)
		0x78	0x01 (Offset Range: 0x01-0x08)
11	INF	0x79	3 (Offset Block No.)
		0x7A	0x0A (Offset Range: 0x0A-0x12)
12	PAM	0x7B	3 (Offset Block No.)
		0x7C	0x14 (Offset Range: 0x14-0X16)
13	FAC	0x7D	3 (Offset Block No.)
		0x7E	0x19 (Offset Range: 0x19-0x1B)
14	CVT	0x7F	3 (Offset Block No.)
		0x80	0x1E (Offset Range: 0x1E-0x20)
15	SUS	0x81	3 (Offset Block No.)
		0x82	0x50 (Offset Range: 0x50--0X55)
16	RES	0x83	3 (Offset Block No.)
		0x84	0x5A (Offset Range: 0x5A-0x5F)
17	CRM	0x85	3 (Offset Block No.)
		0x86	0x03 (Offset Range: 0x32-0x37)

Index	Message	Config Byte	Value
18	CRA	0x87	3 (Offset Block No.)
		0x88	0x03 (Offset Range: 0x3C-0x3E)
19	EXM	0x89	3 (Offset Block No.)
		0x8A	0x03 (Offset Range: 0x41-0x44)
20	FOT	0x8B	3 (Offset Block No.)
		0x8C	0x03 (Offset Range: 0x46-0x48)
21	COT	0x8D	3 (Offset Block No.)
		0x8E	0x03 (Offset Range: 0x23-0x25)
22	REL	0x8F	3 (Offset Block No.)
		0x90	0x01 (Offset Range: 0x96-0x9C)

L3P CIC PPL Event Indications (ANSI)

Event ID	Event
0x01	Reset circuit success indication
0x08	Remote maintenance block or maintenance group block indication
0x09	Remote maintenance unblock or maintenance group unblock indication
0x0A	ACM received
0x0B	ANM received
0x0C	CPG received
0x0E	ISUP protocol violation
0x12	Local maintenance block / group maintenance block success

Event ID	Event
0x13	Local maintenance unblock / group maintenance unblock success
0x15	Local circuit maintenance unblocking indication
0x16	UCIC received
0x17	No GRS/RSC Received, CICs In Service
0x26	CRM received
0x63	DSP Resource Allocation Failure

L3P CIC PPL Event Requests (ANSI)

Event ID	Event
0x01	Send ACM
0x02	Send CPG
0x03	Send ANM
0x04	Request local maintenance blocking (sends BLO)
0x05	Request local maintenance group blocking (sends CGB)
0x06	Request local maintenance unblocking (sends UBL)
0x07	Request local maintenance group unblocking (sends CGU)
0x08	Send GRS
0x09	Send RSC
0x0C	Indicate CPC to wait for incoming COT report when continuity check is being performed on previous circuit.
0x0D	Send incoming continuity check report if continuity check was performed on previous circuit.
0x1F	Send CQM
0x24	Send FAC
0x25	Send PAM
0x2A	Send INR
0x2B	Send INF
0x2D	Send SUS
0x2E	Send RES
0x2F	Send CVT
0x31	IAM Request with parameters
0x40	Manual Stop
0x41	Send Continuity Check Request Message (CCR)
0x45	Send EXM
0x46	Send FOT

L3P CIC PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	Wait For SS7 Network Response	36000
0x02	Wait For L4 Response	3000
0x03	Wait For L4 Maintenance Loopback ACK	3000
0x04	Wait For GRS. Used in OOS Protocol	500

L3P Link (0x0010)

Overview

This section includes all PPL events for the PPL Component L3P Link.

L3P Link PPL Event Indications

Event ID	Event
0x01	Remote Inhibit Confirmation
0x02	Remote Uninhibit Confirmation
0x03	Local Inhibit Denied
0x04	Local Inhibit Timeout
0x05	Local Inhibit Confirmation
0x06	Local Uninhibit Confirmation
0x07	Local Uninhibit Not Possible
0x08	Local Uninhibit Timeout
0x0A	Link In Service
0x0B	Link Out-of-Service, L1_DEAD
0x0C	Link Out-of-Service, L5_OOS3

L3P Link PPL Event Requests

Event ID	Event
0x01	Local Inhibit
0x02	Local Uninhibit
0x03	Set Local Processor Outage
0x04	Clear Local Processor Outage
0x05	Set Busy (Send SIB)
0x06	Clear Busy

ISUP CPC (0x0012)

Purpose This section includes ITU and ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CPC.

Use of Blocks Blocks are used to organize the PPL configuration bytes. Each block does not exceed 200 configuration bytes. The ISUP CPC component has three blocks of configuration bytes. These blocks must be reference when using the bytes.

ISUP CPC Configuration Bytes (ITU)

Block 1

Byte	Description	Value
0x01	Dual-seizure Control Flag	0x00 Drop Outgoing Call 0x01 Drop Incoming Call 0x02 Drop Incoming Call if OPC is less than DPC and CIC is odd, or OPC is greater than DPC and CIC is even (see Q.767 D2.10.1.4) (default)
0x02	Incoming Congestion Control	0x00=Disable 0x01=Enable (default)
0x03	User Part Test Retry Count	4 (default)
0x04	Outgoing SGM required	0=Outgoing SGM not required (default) 1=Outgoing SGM required
Overload Message (OLM)		
0x05	REL/OLM Selection	0x00=Send REL upon detecting incoming congestion (default) 0x01=Send OLM upon detecting incoming congestion
Unrecognizable Parameter Processing		
0x06	Parameter Compatibility Parameter Handling	0x00=PCP Handling Disabled 0x01=PCP Handling Enabled (Default)

Byte	Description	Value
0x07	Unrecognized Parameter Pass On	0x00=Pass On Not Possible 0x01=Pass On Is Possible (Default)
0x08	Reserved	0x00
0x09	Send RFS to host after incoming continuity result is received. When setting this byte to 0x01, the host should not try to send any call processing API on the circuit where incoming Continuity Check is being performed, until continuity check results are received in the form of PPL Event Indication.	0x00 = Send RFS after Continuity has been performed 0x01 = Send RFS before Continuity has been performed
Release (REL) for T8 Expiration		
0x0A	Information Length	0x06
0x0B	Message ID	0x0C (REL)
0x0C	Parameter Count	0x01
0x0D	Parameter 1 ID	0x12 (Cause Indicators)
0x0E	Parameter 1 Data Length	0x02
0x0F	Parameter 1 Data[0]	0x80 (Location=User)
0x10	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
0x11	T7/T9 Timer Expired. Send Event Indication.	0x00=Disabled (default) 0x01=Enabled
Release (REL) for Call Modification Complete (CMC)		
0x12	Information Length	0x06
0x13	Message ID	0x0C (REL)
0x14	Parameter Count	0x01
0x15	Parameter 1 ID	0x12 (Cause Indicators)
0x16	Parameter 1 Length	0x02
0x17	Parameter 1 Data[0]	0x80 (Location=User)
0x18	Parameter 1 Data[1]	0xE6 (Recovery on Timer Expiry)

Byte	Description	Value
Release (REL) for Unexpected RLC Received		
0x19	Information Length	0x06
0x1A	Message ID	0x0C (REL)
0x1B	Parameter Count	0x01
0x1C	Parameter 1 ID	0x12 (Cause Indicators)
0x1D	Parameter 1 Data Length	0x02
0x1E	Parameter 1 Data[0]	0x80 (Location=User)
0x1F	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
Release (REL) for Normal L3P		
0x20	Information Length	0x06
0x21	Message ID	0x0C (REL)
0x22	Parameter Count	0x01
0x23	Parameter 1 ID	0x12 (Cause Indicators)
0x24	Parameter 1 Length	0x02
0x25	Parameter 1 Data[0]	0x80 (Location=User)
0x26	Parameter 1 Data[1]	0x90 (Normal Release)
0x27	Unused	
Release (REL) for BLK in Wait for ACM State		
0x28	Information Length	0x06
0x29	Message ID	0x0C (REL)
0x2A	Parameter Count	0x01
0x2B	Parameter 1 ID	0x12 (Cause Indicators)
0x2C	Parameter 1 Data Length	0x02
0x2D	Parameter 1 Data[0]	0x80 (Location=User)
0x2E	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
0x2F–0x36	Unused	
Release (REL) for Congestion		
0x37	Information Length	0x09

Byte	Description	Value
0x38	Message ID	0x0C (REL)
0x39	Number of default parameters	0x02
0x3A	Parameter 1 ID	0x12 (Cause Indicators)
0x3B	Parameter 1 Data Length	0x02
0x3C	Parameter 1 data[0]	0x80 (Location=User)
0x3D	Parameter 1 data[1]	0xAA (Cause 42=Resource Unavailable, Switching Equipment Congestion)
0x3E	Parameter 2 ID	0x27 (Automatic Congestion Level)
0x3F	Parameter 2 Data Length	0x01
0x40	Parameter 2 Data[0]	0x02 (Automatic Congestion Level 2)
0x41–0x45	Unused	
Release (REL) for T7 Expiration		
0x46	Information Length	0x06
0x47	Message ID	0x0C (REL)
0x48	Number of default parameters	0x01
0x49	Parameter 1 ID	0x12 (Cause Indicators)
0x4A	Parameter 1 Data Length	0x02
0x4B	Parameter 1 Data[0]	0x80 (Location=User)
0x4C	Parameter 1 Data[1]	0xA9 (Temp Failure)
Release (REL) for T9 Expiration		
0x55	Information Length	0x06
0x56	Message ID	0x0C (REL)
0x57	Parameter Count	0x01
0x58	Parameter ID	0x12 (Cause Indicators)
0x59	Parameter 1 Length	0x02
0x58	Parameter 1 Data[0]	0x80 (Location=User)
0x59	Parameter 1 Data[1]	0xA9 (Temp Failure)
Call Modification Reject (CMRJ)		
0x5C	Information Length	0x05
0x5D	Message ID	0x1E (CMRJ)

Byte	Description	Value
0x5E	Parameter Count	0x01
0x5F	Parameter 1 ID	0x17 (Call Modification Indicator)
0x60	Parameter 1 Length	0x01
0x61	Parameter 1 Data[0]	0x01
Continuity (COT)		
0x64	Information Length	0x04
0x65	Message ID	0x05 (COT)
0x66	Parameter Count	0x01
0x67	Parameter ID	0x12 (Cause Indicators)
0x69	Parameter Length	0x00
Release (REL) for T33 Expiration		
0x6E	Information Length	0x06
0x6F	Message ID	0x0C (REL)
0x70	Parameter Count	0x01
0x71	Parameter 1 ID	0x12 (Cause Indicators)
0x72	Parameter 1 Length	0x02
0x73	Parameter 1 Data[0]	0x80 (Location=User)
0x74	Parameter 1 Data[1]	0x9F (Normal Unspecified)
Release (REL) for T2 Expiration		
0x82	Length of the data	0x06
0x83	Message ID	0x0C MCP/PCP
0x84	Parameter Count	0x01
0x85	Parameter 1 ID	0x12 (Cause Indicators)
0x86	Parameter 1 Length	0x02
0x87	Parameter 1 Data[0]	0x80 (Location=User)
0x88	Parameter 1 Data[1	0xE6 (Cause 102=Protocol Error, Recovery on Timer Expiration)
Release (REL) for PCP		

Byte	Description	Value
0x8C	Information Length	0x06
0x8D	Message ID	0x0C (REL)
0x8E	Parameter Count	0x01
0x8F	Parameter 1 ID	0x12 (Cause Indicators)
0x90	Parameter 1 Length	0x02
0x91	Parameter 1 Data [0]	0x80 (Location=User)
0x92	Parameter 1 Data [1]	0xE3 (Cause 99=Protocol Error, IE or Parameter Not Implemented, Discarded)
Confusion (CFN) for PCP		
0x96	Information Length	0x06
0x97	Message ID	0x2F (CFN)
0x98	Parameter Count	0x01
0x99	Parameter 1 ID	0x12 (Cause Indicators)
0x9A	Parameter 1 Length	0x02
0x9B	Parameter 1 Data[0]	0x80
0x9C	Parameter 1 Data[1]	0xE3 (Cause 99=Protocol Error, IE or Parameter Not Implemented, Discarded)
Confusion (CFN) with Unrecognized Parameters		
0xA0	Information Length	0x06
0xA1	Message ID	0x2F (CFN)
0xA2	Parameter Count	0x01
0xA3	Parameter 1 ID	0x12 (Cause Indicators)
0xA4	Parameter 1 Length	0x02
0xA5	Parameter 1 Data [0]	0x80 (Location=User)
0xA6	Parameter 1 Data [1]	0xEE (Cause 110=Protocol Error, Message with Unrecognized Parameters, Discarded)
Facility Reject (FRJ) for PCP		
0xAA	Information Length	0x09
0xAB	Message ID	0x21 (FRJ)
0xAC	Parameter Count	0x02

Byte	Description	Value
0xAD	Parameter 1 ID	0x18 (Facility Indicator)
0xAE	Parameter 1 Length	0x01
0xAF	Parameter 1 Data [0]	0x02 (User-to-User Service)
0xB0	Parameter 2 ID	0x12 (Cause Indicators)
0xB1	Parameter 2 Length	0x02
0xB2	Parameter 2 Data [0]	0x80 (Location=User)
0xB3	Parameter 2 Data [1]	0xE3 (Cause 99=Protocol Error, IE or Parameter Not Implemented, Discarded)
Release Complete (RLC) for PCP		
0xB4	Information Length	0x06
0xB5	Message ID	0x10 (RLC)
0xB6	Parameter Count	0x01
0xB7	Parameter 1 ID	0x12 (Cause Indicators)
0xB8	Parameter 1 Length	0x02
0xB9	Parameter 1 Data[0]	0x80 (Location=User)
0xBA	Parameter 1 Data [1]	0xE3 (Cause 99=Protocol Error, IE or Parameter Not Implemented, Discarded)
0xBB	Length of the Data Structure	0x06
User Part Available (UPA)		
0xBE	Information Length	0x06
0xBF	Message ID	0x35 (UPA)
0xC0	Parameter Count	0x01
0xC1	Parameter 1 ID	0x39 (Parameter Compatibility Information)
0xC2	Parameter 1 Length	0x02
0xC3	Parameter 1 Data[0]	0x00

Byte	Description	Value
0xC4	Parameter Data[1]	0x00 (End Node Interp, Do Not Release, Do Not Notify, Discard Message, Discard Parameter)

Block 2

Value	Description	Value
Miscellaneous Information		
0x01	In-call Modification	0=Not Allowed 1=Allowed (default)
0x02	Release call on Timeout	0=Don't Release Call 1=Release Call (default)
0x03	Continuity Control	0=In-switch Continuity enabled (default) 1=Host Controlled Continuity enabled
0x0A	Sending SAM to Host/L3P CIC Note: If you change this byte to 0x01, you must also set component 0x000F, byte 0x30, Block 0x01 to value 0x01	0x00=Send to Host (default) 0x01=Send to L3P CIC

ISUP CPC PPL Event Indications (ITU)

Event ID	Event
0x01	Continuity indication in IAM. (**NOTE:** The host also receives a DS0 status change as the channel is automatically brought OOS to perform the continuity test.)
0x02	Continuity Check Incoming Success

Event ID	Event
0x03	Continuity Check Incoming Failure
0x04	T8 expiration
0x05	REL received while waiting for Continuity Check
0x06	Alert Maintenance=No RLC received before T5 expiration
0x07	Continuity Check Outgoing Success
0x08	Continuity Check Outgoing Failure
0x09	Extra USR message received in UUS2
0x0B	Invalid PAM request received from host
0x0C	Only user-initiated SUS accepted
0x0D	Suspended from terminating side, but received resume from originating side
0x0E	Suspended from originating side but received resume from terminating side
0x0F	Only user-initiated RES accepted
0x11	Max USR messages sent
0x12	T39 expiry
0x13	RES not received OPR timeout
0x14	Maximum number of UPT retries completed without response
0x17	Discarded message with unrecognized parameters
0x20	FAR received
0x21	FAA received
0x22	FRJ received
0x23	USR received
0x24	FAC received
0x25	PAM received
0x26	NRM received
0x27	CRG received

Event ID	Event
0x28	IDR received
0x29	IRS received
0x2A	INR received
0x2B	INF received
0x2F	SUS received
0x30	RES received
0x32	SAM received
0x34	UPA received
0x38	FOT received
0x39	DRS received
0x3A	TDRS expiry
0x41	MPM received indication
0x42	CCL received indication
0x43	OPR received
0x44	OPR resume received
0x45	UPT message indication
0x49	CMC received
0x4A	CMRJ received
0x4B	CMR received
0x4C	TCMR expiry
0x4D	In call modification not allowed
0x4E	Parameter discarded (Raw SS7 Data Parameters ICB 0x22 included in this event indication)
0x4F	Indication with data for CCR
0x50	Indication with data for LPA
0x55	Loop Prevention (LOP).
0x99	T9 expiration
0x9A	T7 expiration

ISUP CPC PPL Timers (ITU)

Timer ID	Timer	Default Value (10ms)
0x01	T8	1500
0x02	T5	36000
0x03	T1	1500
0x04	T7	2000
0x05	T9	18000
0x06	T33	1500
0x07	T39	1200
0x08	T6	3000
0x09	T35	1500
0x0A	T2	18000
0x0B	T4:UPT re-try timer (only for ITU)	36000
0x0C	T3 for OLM message	12000
0x0D	Tres for Circuit Reservation ACK	18000
0x0E	Tdrs for DRS message	18000

ISUP CPC Configuration Bytes (ANSI)

Byte	Description	Value
0x0 1	Dual-seizure Control Flag	0x00 Drop Outgoing Call 0x01 Drop Incoming Call 0x02 Drop Incoming Call if OPC is less than DPC and CIC is odd, or OPC is greater than DPC and CIC is even (see Q.767 D2.10.1.4) [default]
0x02	Incoming Congestion Control	0x00=Disable 0x01=Enable (default)
0x03 – 0x08	Unused	

Byte	Description	Value
0x09	Send RFS to host after incoming continuity result is received. When setting this byte to 0x01, the host should not try to send any call processing API on the circuit where incoming Continuity Check is being performed, until continuity check results are received in the form of PPL Event Indication.	0x00 = Send RFS after Continuity has been performed 0x01 = Send RFS before Continuity has been performed
Release (REL) for T8 Expiration		
0x0A	Information Length	0x06
0x0B	Message ID	0x0C (REL)
0x0C	Parameter Count	0x01
0x0D	Parameter 1 ID	0x12 (Cause Indicators)
0x0E	Parameter 1 Data Length	0x02
0x0F	Parameter 1 Data[0]	0x80 (Location=User)
0x10	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
0x11	T7/T9 Timer Expired. Send Event Indication.	0x00=Disabled (default) 0x01=Enabled
0x12 – 0x18	Unused	0x00
Release (REL) for Unexpected RLC Received		
0x19	Information Length	0x06
0x1A	Message ID	0x0C (REL)
0x1B	Parameter Count	0x01
0x1C	Parameter 1 ID	0x12 (Cause Indicators)
0x1D	Parameter 1 Data Length	0x02
0x1E	Parameter 1 Data[0]	0x80 (Location=User)
0x1F	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
Release (REL) Normal L3P Release		

Byte	Description	Value
0x20	Information Length	0x06
0x21	Message ID	0x0C (REL)
0x22	Parameter Count	0x01
0x23	Parameter 1 ID	0x12 (Cause Indicators)
0x24	Parameter 1 Length	0x02
0x25	Parameter 1 Data[0]	0x80 (Location=User)
0x26	Parameter 1 Data[1]	0x90 (Normal Release)
0x27	Unused	
Release (REL) for BLK in Wait-for-ACM State		
0x28	Information Length	0x06
0x29	Message ID	0x0C (REL)
0x2A	Parameter Count	0x01
0x2B	Parameter 1 ID	0x12 (Cause Indicators)
0x2C	Parameter 1 Data Length	0x02
0x2D	Parameter 1 Data[0]	0x80 (Location=User)
0x2E	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
0x2F – 0x36	Unused	
Release (REL) for T7 Expiration		
0x37	Information Length	0x06
0x38	Message ID	0x0C (REL)
0x39	Parameter Count	0x01
0x3A	Parameter 1 ID	0x12 (Cause Indicators)
0x3B	Parameter 1 Data Length	0x02
0x3C	Parameter 1 Data[0]	0x80 (Location=User)
0x3D	Parameter 1 Data[1]	0xA9 (Cause 41=Resource Unavailable, Temp Failure)
0x3E-0x45	Unused	
Release (REL) (Temp Failure) on Congestion Parameters		
0x46	Information Length	0x09

Byte	Description	Value
0x47	Message ID	0x0C (REL)
0x48	Number of default parameters	0x02
0x49	Parameter 1 ID	0x12
0x4A	Parameter 1 Data Length	0x02
0x4B	Parameter 1 Data[0]	0x80 (Location=User)
0x4C	Parameter 1 Data[1]	0xAA
0x4D	Parameter 2 ID	0x27
0x4E	Parameter 2 Data Length	0x01
0x4F	Parameter 2 Data[0]	0x02
0x50-0x54	Unused or User-defined	
Release (REL) for T9 Expiration		
0x55	Information Length	0x06
0x56	Message ID	0x0C (REL)
0x57	Parameter Count	0x01
0x58	Parameter 1 ID	0x12 (Cause Indicators)
0x59	Parameter 1 Length	0x02
0x58	Parameter 1 Data[0]	0x80 (Location=User)
0x59	Parameter 1 Data[1]	0xA9 (Temp Failure)
Continuity (COT)		
0x64	Information Length	0x04
0x65	Message ID	0x05 (COT)
0x66	Parameter Count	0x01
0x67	Parameter 1 ID	0x10 (Continuity Indicators)
0x69	Parameter 1 Length	0x00
Release (REL) for T33 Expiration		
0x6E	Information Length	0x06
0x6F	Message ID	0x0C (REL)
0x70	Parameter Count	0x01
0x71	Parameter 1 ID	0x12 (Cause Indicators)

Byte	Description	Value
0x72	Parameter 1 Length	0x02
0x73	Parameter 1 Data[0]	0x80 (Location=User)
0x74	Parameter 1 Data[1]	0x9F (Normal Unspecified)
Release (REL) for T6 Expiration		
0x82	Information Length	0x06
0x83	Message ID	0x0C (REL)
0x84	Parameter Count	0x01
0x85	Parameter 1 ID	0x12 (Cause Indicators)
0x86	Parameter 1 Length	0x02
0x87	Parameter 1 Data[0]	0x80 (Location=User)
ANSI ISUP Segmentation		
0xD4	ANSI ISUP Segmentation	0x00 E bit in forward indicator is treated as a spare. No ANSI segmentation. (default) 0x01 E bit in forward indicator is for ANSI segmentation

ISUP CPC PPL Event Indications (ANSI)

Event ID	Event
0x01	Continuity indication in IAM. (**NOTE:** The host will also receive a DS0 status change as the channel is automatically brought OOS to perform the continuity test.)
0x02	COT success
0x03	COT failure
0x04	T8 expiration
0x05	REL received while waiting for Continuity Check
0x06	Alert Maintenance=No RLC received before T5 expiration
0x07	Continuity Check Outgoing Success
0x08	Continuity Check Outgoing Failure
0x0A	Only network-initiated SUS accepted
0x0B	Invalid PAM request received from host

Event ID	Event
0x0C	TCRA expiry
0x0D	TCRM expiry
0x0E	CRA received
0x10	Only network-initiated RES accepted.
0x17	Discarded message with unrecognized parameters.
0x20	FAR received
0x21	FAA received
0x24	FAC received
0x25	PAM received
0x2A	INR received
0x2B	INF received
0x2F	SUS received
0x30	RES received
0x37	EXM received
0x38	FOT received
0x4E	Discarded parameter (Raw SS7 Data Parameters ICB 0x22 included in this event indication)
0x4F	CCR received
0x50	LPA received
0x55	Loop Prevention (LOP).
0x99	T9 expiration
0x9A	T7 expiration

ISUP CPC PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	T8	1500
0x02	T5	6000
0x03	T1	1500
0x04	T7	2000

Timer ID	Timer	Default Value (10ms)
0x05	T9	18000
0x06	T33	1500
0x07	T6	3000
0x08	TCRA	1000
0x09	TCRM	1000
0x0D	T36	200

ISUP SPRC (0x0013)

Purpose

This section includes ITU and ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP SPRC.

ISUP SPRC Configuration Bytes (ITU)

You can modify the parameters for Config Bytes 1–4, but you cannot move them to other byte locations. Other functions in the system access these parameters.

Byte	Description	Value
0x01	Parameter ID	0x16 (Range and Status)
0x02	Byte offset of range field	0x00
0x03	Service Indicator	0x05
0x04	Subservice Field	0x02
0x05	MTP Pause Logic Flag	0x00 Disable [default] 0x01 Enable
0x06–0x09	Unused	
Confusion (CFN) for MCP		
0x0A	Information Length	0x06
0x0B	Message ID	0x2F (CFN)
0x0C	Parameter Count	0x01
0x0D	Parameter 1 ID	0x12 (Cause Indicators)
0x0E	Parameter 1 Length	0x02
0x0F	Parameter 1 Data[0]	0x80 (Location=User)
0x10	Parameter 1 Data[1]	0xE1 (Cause 97=Protocol Error, Message Type Not Implemented)
0x11–0x13	Not Used	
0x14	Information Length	0x06
0x15	Message ID	0x2F (CFN)
0x16	Parameter Count	0x01
0x17	Parameter 1 ID	0x12 (Cause Indicators)
0x18	Parameter 1 Length	0x02

Byte	Description	Value
0x19	Parameter 1 Data[0]	0x80 (Location=User)
0x1A	Parameter 1 Data[1]	0xEE (Cause 110=Protocol Error, Message with Unrecognized Parameters, Discarded)
Unequipped CIC (UCIC)		
0x1E	Information Length	0x02
0x1F	Message ID	0x2E (UCIC)
0x20	Parameter Count	0x00
0x21–0x27	Not Used	
0x28	Enable/Disable ACC Local Overload Algorithm	0x01=Enabled 0x02=Queue Probe
0x29	Outstanding QueueProbe Count for ACL0 Low Level	0x00
0x2A	Outstanding QueueProbe Count for ACL0 High Level	0x04
0x2B	Outstanding QueueProbe Count for ACL1 High Level	0x03
0x2C	Outstanding QueueProbe Count for ACL1 High Level	0x0B
0x2D	Outstanding QueueProbe Count for ACL2 Low Level	0x09
0x2E	Outstanding QueueProbe Count for ACL2 Low Level	0x0B
0x2F	Congestion Cleared Threshold for Outgoing Calls, 100 ms units	10 ms 0x0A
0x30	Number of probes to determine congestion cleared=Outgoing Calls	10 0x0A
0x31	Defines the range of channels in GRS	0x0C (Default value or indicates maximum number of CICs per span)

Byte	Description	Value
0x32	Defines whether purging is on or off in the case of CCS redundancy loss.	0x01 Enabled
0x3C	Exchange Type	0x00=Type A 0x01=Type B (default)
Unrecognized Message Processing		
0x3D	Message Compatibility Parameter (MCP) Processing Notification	0x00=Do Not send PPL Event to Host 0x01=Send PPL Event to Host
0x3E	Unrecognized Parameter Pass On	0x00=Pass On Not Possible 0x01=Pass On Is Possible
Release (REL) for Message Not Supported		
0x42	CGRS stop upon receipt of UCIC	0x00 = Stop 0x01 = Do not stop (Default)
0x46	Information Length	0x06
0x47	Message ID	0x0C (REL)
0x48	Parameter Count	0x01
0x49	Parameter 1 ID	0x12 (Cause Indicators)
0x4A	Parameter 1 Length	0x02
0x4B	Parameter 1 Data [0]	0x80 (Location=User)
0x4C	Parameter 1 Data [1]	0xE1 (Cause 97=Message Type Not Implemented)
ISUP Automatic Congestion Control (ACC)		
0x50	Multiplicative Decrement Constant Beta_0 normalized to 128	0x80
0x51	Multiplicative Decrement Constant Beta_1	0x80
0x52	Multiplicative Decrement Constant Beta_2	0x80

Byte	Description	Value
0x53	Multiplicative Decrement Constant Beta_3	0x80
0x54	Multiplicative Decrement Constant Beta_4	0x7A
0x55	Multiplicative Decrement Constant Beta_5	0x76
0x56	Multiplicative Decrement Constant Beta_6	0x6E
0x57	Multiplicative Decrement Constant Beta_7	0x64
0x58	Multiplicative Decrement Constant Beta_8	0x5A
0x59	Multiplicative Decrement Constant Beta_9	0x5A
0x5A	Multiplicative Decrement Constant Beta_10	0x50
0x5B	Multiplicative Decrement Constant Beta_11	0x32
0x5C	Multiplicative Decrement Constant Beta_12	0x28
0x5D	Multiplicative Decrement Constant Beta_13	0x1E
0x5E	Multiplicative Decrement Constant Beta_14	0x0A
0x5F	Multiplicative Decrement Constant Beta_15	0x00
0x60	Additive Increase Constant - Alpha	0x02
0x61	Maximum Residual delay of a Queue Probe	0x64
0x62	Accuracy Limit	0x80
0x90	ACL0 Low Level Offset for CPU Utilization	0x5A
0x91	ACL0 High Level Offset for CPU Utilization	0x55

Byte	Description	Value
0x92	ACL1 Low Level Offset for CPU Utilization	0x53
0x93	ACL1 High Level Offset for CPU Utilization	0x5B
0x94	ACL2 Low Level Offset for CPU Utilization	0x58
0x95	ACL2 High Level Offset for CPU Utilization	0x5F
0x8C	Type of STACK	0x00=Non-BT Stack(default) 0x01=BT Stack
0x8D	CS ACL2 Fraction Allowed	0x20
Release (REL) for INF Timer Expiration		
0x6E	Information Length	0x06
0x6F	Message ID	0x0C (REL)
0x70	Parameter Count	0x01
0x71	Parameter 1 ID	0x12
0x72	Parameter 1 Length	0x02
0x73	Parameter 1 Data [0]	0x80 (Location=User)
0x74	Parameter 1 Data [1]	0x9F (Cause 31=Normal Unspecified)
0x75–0xC8	Unused	

ISUP SPRC PPL Event Indications (ITU)

Event ID	Event
0x01	Format error in incoming message (**NOTE**: The PPL Event Indication has AIB of stack)
0x02	Unequipped CIC in incoming message (**NOTE:** The PPL Event Indication has an AIB of stack)
0x03	Undefined message received (**NOTE:** The PPL Event Indication has an AIB of stack)
0x04	Confusion Message received (**NOTE:** The PPL Event Indication will have a AIB of span/channel)
0x05	Remote User Part Available (**NOTE**: The PPL Event Indication has AIB of channel)
0x06	Pass On unrecognized/unexpected message
0x07	Unrecognized message received on unequipped CIC
0x08	Discard message with unresolved parameters.

ISUP SPRC PPL Timers (ITU)

Timer ID	Timer	Default Value (10ms)
0x01	MTP Pause timer base (pause expiration is 10 times this value)	400
0x02	Queue Probe Timer	10

ISUP SPRC Configuration Bytes (ANSI)

You can modify the parameters for Config Bytes 1–4, but you cannot move them to other byte locations. Other functions in the system access these parameters.

Byte	Description	Value
0x01	Parameter ID	0x16 (Change and Status)
0x02	Byte offset of range field	0x00
0x03	ANSI Service Indicator	0x05
0x04	ANSI Network Indicator	0x02

Byte	Description	Value
0x05	MTP Pause Logic Flag	0x00 Disable [default] 0x01 Enable
Confusion (CFN) for Message Type Unknown		
0x06	Information Length	0x06
0x07	Message ID	0x2F (CFN)
0x08	Parameter Count	0x01
0x09	Parameter 1 ID	0x12 (Cause Indicators)
0x0A	Parameter 1 Length	0x02
0x0B	Parameter 1 Data[0]	0x80 (Location=User)
0x0C	Parameter 1 Data[1]	0xE1
0x0D-0F	Not Used	
Confusion (CFN) for Protocol Error		
0x14	Information Length	0x06
0x15	Message ID	0x2F (CFN)
0x16	Parameter Count	0x01
0x17	Parameter 1 ID	0x12 (Cause Indicators)
0x18	Parameter 1 Length	0x02
0x19	Parameter 1 Data[0]	0x80 (Location=User)
0x1A	Parameter 1 Data[1]	0xEF
0x1B–0x1D	Unused	0x00
Unequipped CIC (UCIC)		
0x1E	Information Length	0x02
0x1F	Message ID	0x2E
0x20	Parameter Count	0x00
0x21–0x27	Unused	
0x28	Number of probes to determine Average Delay	10
0x29	Probe Delay Threshold for Incoming Calls, 100 ms units	100 ms

Byte	Description	Value
0x2A	Average Delay Threshold for Incoming Calls, 100 ms units	50 ms
0x2B	Congestion Cleared Threshold for Incoming Calls, 100 ms units	10 ms
0x2C	Number of probes to determine congestion cleared=Incoming Calls	10
0x2D	Probe Delay Threshold for Outgoing Calls, 100 ms units	100 ms
0x2E	Average Delay Threshold for Outgoing Calls, 100 ms units	50 (5 s)
0x2F	Congestion Cleared Threshold for Outgoing Calls, 100 ms units	10 (1 s)
0x30	Number of probes to determine congestion cleared=Outgoing Calls	10
0x31	Defines the range of channels in GRS	0xFF (Default value or indicates maximum number of CICs per span)
0x32	Defines whether purging is on or off in the case of CCS redundancy loss.	0x01 Enabled
Circuit Validation Response (CVR)		
0x32	Information Length	0x08
0x33	Message ID	0xEB
0x34	Parameter Count	0x02
0x35	Parameter 1 ID	0xE6 (CVR)
0x36	Parameter 1 Length	0x01
0x37	Parameter 1 Data[0]	0x01
0x38	Parameter 2 ID	0xE5 (CGC)
0x39	Parameter 2 Data Length	0x01
0x3A	Parameter 2 Data[0]	0x01
0x3C	Exchange Type	0x00=Type A 0x01=Type B (default)
0x3D–0x6D	Not Used	

Byte	Description	Value
Release (REL) (INF Timer Expiry)		
0x6E	Information length	0x06
0x6F	Message ID	0x0C
0x70	Parameter Count	0x01
0x71	Parameter 1 ID	0x12
0x72	Parameter 1 Length	0x02
0x73	Parameter 1 Data [0]	0x80 (Location=User)
0x74	Parameter 1 Data [1]	0x9F (Temp Failure)
0x75–0xC8	Unused or User-defined	

ISUP SPRC PPL Event Indications (ANSI)

Event ID	Event
0x01	Format error in incoming message (**NOTE:** The PPL Event Indication will have a AIB of stack)
0x02	Unequipped CIC in incoming message (**NOTE:** The PPL Event Indication has an AIB of stack unless the incoming DPC is not configured. In that case, a Channel AIB with data of 0xFF FF FF is used.)
0x03	Undefined message received (**NOTE:** The PPL Event Indication will have a AIB of stack)
0x04	Confusion Message received (**NOTE:** The PPL Event Indication will have a AIB of span/channel)

ISUP SPRC PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	MTP Pause timer base (pause expiration is 10 times this value)	400

Timer ID	Timer	Default Value (10ms)
0x02	Time between probes for Congestion Control	100

ISUP CQS (0x0076)

Purpose This section includes ITU and ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CQS.

ISUP CQS Configuration Bytes (ITU)

Byte	Description	Value
0x01	Max Range	31
0x02	Reserved	0x00
0x03	Circuit State Indicator Parameter ID	0x26 (Circuit State Indicator)
Release (REL) for Protocol Error		
0x03	Information Length	0x26
0x0B	Message ID	0x0C (REL)
0x0C	Parameter Count	0x01
0x0D	Parameter 1 ID	0x12 (Cause Indicators)
0x0E	Parameter 1 Data Length	0x02
0x0F	Parameter 1 Data[0]	0x80 (Location=User)
0x10	Parameter 1 Data[1]	0xEF (Cause 111=Protocol Error, Unspecified)

ISUP CQS PPL Event Indications (ITU)

Event ID	Event
0x02	CQR Indication
0x01	Timer T28 Expiration

ISUP CQS PPL Timers (ITU)

Timer ID	Timer	Default Value (10ms)
0x01	T28	1000

ISUP CQS Configuration Bytes (ANSI)

Byte	Description	Value
0x01	Max Range	31
0x02	Recovery Action (as defined by ANSI T1.113) should be taken if this byte is set to 0x01.	0x00=Do Not Perform Auto-Recovery 0x01=Perform Auto-Recovery
0x03	Circuit State Indicator Parameter ID	0x26 (Circuit State Indicator)
Release (REL) for Protocol Error		
0x0A	Information Length	0x06
0x0B	Message ID	0x0C (REL)
0x0C	Parameter Count	0x01
0x0D	Parameter 1 ID	0x12 (Cause Indicators)
0x0E	Parameter 1 Data Length	0x02
0x0F	Parameter 1 Data[0]	0x80 (Location=User)
0x10	Parameter 1 Data[1]	0xEF (Cause 111=Protocol Error, Unspecified)

ISUP CQR (0x0077)

Purpose This section includes ITU and ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CQR.

ISUP CQR Configuration Bytes (ITU)

Byte	Description	Value
0x01	Message ID	0x2B (CQR)
0x02	Max Range	31
0x03	Circuit state indicator parameter ID	0x26
0x04	CQM Indication to host	0x00=Disable (default) 0x01=Enable

ISUP CQR Configuration Bytes (ANSI)

Byte	Description	Value
0x01	Message ID	0x2B (CQR)
0x02	Max Range	31
0x03	Circuit state indicator parameter ID	0x26
0x04	CQM Indication to host	0x00=Disable (default) 0x01=Enable

ISUP CQR PPL Event Indications (ITU)

Event ID	Event
0x01	CQM indication with SS7 ICB

ISUP CQR PPL Event Indications (ANSI)

Event ID	Event
0x01	CQM received

ISUP CVS (0x0078)

Purpose This section includes ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CVS.

ISUP CVS Configuration Bytes (ANSI) For CLLI and CIN parameter formats, refer to ANSI T1-113.

Byte	Description	Value
0x01	CIN	0x00=Not Configured 0x01=Configured (default)
0x02	CLLI	0x00=Not configured 0x01=Configured (default)
0x03	CVR Processing	0x00=Do Not Process/Send PPL Event (default) 0x08 (CVR Received) to Host 0x01=Send CVR
0x04 (Continued on next page)	CGC Indicator	0x00 (default) Bits 1, 0=Circuit Group Carrier Indicator 00=Unknown 01=Analog 10=Digital 11=Digital and Analog. Bits 3, 2=Double Seizing Indicator 00=Always 00; configuration is taken from CPC data. **DO NOT CHANGE**.

Byte	Description	Value
0x04 (Continued)	CGC Indicator	Bits 5, 4=Alarm Carrier Indicator 00=Unknown 01=Software Carrier Handling 10=Hardware Carrier Handling 11=Spare Bits 7, 6- Continuity Check Requirement Indicator 00=Unknown; 01=None; 10=Statistical; 11=Per Call
0x05-0x06	Unused	
0x07	Timer to be used by UPU/CVT Test	0–24 Timer 2 (30s) (default)
0x08	CVT Re-attempt Count	0–255 8 (default)
0x09	Unused	

ISUP CVS PPL Event Indications (ANSI)

Event ID	Event
0x01	CVT Successful=CIN Match, CGC Match
0x02	CVT Successful=CIN Mismatch Data[0–25]=Local CIN Data[26–51]=Remote CIN
0x03	CVT Successful=CGC Data Inconsistent Data[0]=Remote CGC Parameter Data[1–26]=Local CIN

Event ID	Event
0x04	CVT Failure=CLLI Not Received
0x05	CVT Failure=CLLI Received Data[0–10]=Remote CLLI
0x06	CVR Receive Failed after Reattempt
0x07	Local Circuit Translation Data Absent
0x08	CVR Received=CVR in TLV Format Data [0]=Message Type [CVR] Data [1]=Number of Parameters Data [2]=CVR Indicator Parameters Code Data [3]=CVR Indicator Length Data [4]=CVR Indicator Value Data [5]=CGC Parameters Code Data [6]=CGC Parameters Length Data [7]=CGC Parameters Value Data [8-n]=Optional Parameters [TLV]

ISUP CVS PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	TCVT	1000
0x02	T37	3000

ISUP CVR (0x0079)

Purpose

This section includes ANSI PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CVR.

ISUP CVR Configuration Bytes (ANSI)

For the CLLI and CIN parameter formats in the table below, refer to ANSI T1-113.

Byte	Description	Value
0x01	CLII	0x00=Not Configured 0x01=Configured (default)
0x02	CIN	0x00=Not configured 0x01=Configured (default)
0x03	CGC Indicator	0x00 (default) Bits 1, 0=Circuit Group Carrier Indicator 00=Unknown 01=Analog 10=Digital 11=Digital and Analog. Bits 3, 2=Double Seizing Indicator 00=Always 00; configuration taken from CPC data. **DO NOT CHANGE** Bits 5, 4=Alarm Carrier Indicator 00=Unknown 01=Software Carrier Handling 10=Hardware Carrier Handling 11=Spare Bits 7, 6- Continuity Check Requirement Indicator 00=Unknown; 01=None; 10=Statistical; 11=Per Call

Byte	Description	Value
0x04	Message ID	0xEB (CVR)
0x05	Number of Mandatory Parameters	2 (default)
0x06	CVR Indicator Parameter Code	0xE6
0x07	CVR Indicator Parameter Length	0x01
0x08	CGC Parameter Code	0xE5
0x09	CGC Parameter Length	0x01
0x55	CVT Indication to host	0x00=Disable (default) 0x01=Enable

ISUP CVR PPL Event Indications (ANSI)

Event ID	Event
0x01	CVT received

ISUP CCO (0x0080)

Purpose

This section includes ITU and ANSI PPL Config Bytes and Timer information for the PPL Component ISUP CCO.

ISUP CCO Configuration Bytes (ITU/ANSI)

Bytes 1–5 define the format of the COT message. These should not be changed.

Byte	Description	Value
0x01	Information Length	0x04
0x02	Message ID	0x05 (COT)
0x03	Parameter Count	0x01
0x04	Parameter 1 ID	0x10 (Continuity Indicators)
0x05	Parameter 1 Data Length	0x00

ISUP CCO PPL Timers (ITU/ANSI)

Timer ID	Timer	Default Value (10ms)
0x18	T24	200

ISUP CRCS (0x0081)

Purpose This section includes (ITU only) PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CRCS.

ISUP CRCS Configuration Bytes (ITU) Bytes 1–5 define the format of the COT message. These should not be changed.

Byte	Description	Value
0x01	Information Length	0x04
0x02	Message ID	0x05 (COT)
0x03	Parameter Count	0x01
0x04	Parameter1 ID	0x10 (Continuity Indicator)
0x05	Parameter 1 Data Length	0x00
0x06–0x09	Not Used	
0x0A	Continuity Recheck	0x00=Enables unlimited repeated continuity checks 0x01= Limits the repeated continuity checks to n+2 times where N is the value of config byte 0x0B. (default)
0x0B	Number of times to recheck	2 (default)

Release (REL) when COT successful		
0x14	Information Length	0x06
0x15	Message ID	0x0C (REL)
0x16	Number of Parameters	0x01

0x17	Parameter 1 Data Type	0x12 (Cause Indicators)
0x18	Parameter 1 Data Length	0x02 (Cause Indicators)
0x19	Parameter 1 Data[0]	0x80 (Location=User)
0x1A	Parameter 1 Data[1]	0x90 (Cause 16=Normal Call Clearing)

Release (REL) if RLC is not received		
0x1E	Information Length	0x06
0x1F	Message ID	0x0C (REL)
0x20	Number of Parameters	0x01
0x21	Parameter 1 Data Type	0x12 (Cause Indicators)
0x22	Parameter 1 Data Length	0x02 (Cause Indicators)
0x23	Parameter 2 Data [0]	0x80 (Location=User)
0x24	Parameter 1 Data[1]	0xE6 (Cause 102; Recovery on timer expiry)

ISUP CRCS PPL Event Indications (ITU)

Event ID	Event
0x01	T36 expiry on Continuity Recheck
0x02	COT failure indication (dual seizure), process network IAM
0x03	T5 expiry in waiting for RLC state
0x04	CPC Not IDLE Indication
0x05	Release complete received from network

Event ID	Event
0x06	COT success indication

ISUP CRCS PPL Timers (ITU)

Timer ID	Timer	Default Value (10ms)
0x01	T1	1500
0x05	T5	30000
0x18	T24	200
0x19	T25	1000
0x1A	T26	18000

ISUP DCO (0x0082)

Purpose

This section includes ANSI (only) PPL Config Bytes and Timer information for the PPL Component ISUP DCO.

ISUP DCO Configuration Bytes (ANSI)

Config Bytes 1–5 define the format of the COT message. These should not be changed.

Byte	Description	Value
0x01	Information Length	0x04
0x02	Message ID	0x05 (COT)
0x03	Parameter Count	0x01
0x04	Parameter 1 ID	0x10 (Continuity Indicators)
0x05	Parameter 1 Data Length	0x00

ISUP DCO PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	TCCR	200
0x18	T24	200

ISUP CRO (0x0083)

Purpose

This section includes ANSI (only) PPL Config Bytes, all PPL Events, and Timer information for the PPL Component ISUP CRO.

ISUP CRO Configuration Bytes (ANSI)

Bytes 1–5 define the format of the COT message. These should not be changed.

Byte	Description	Value
0x01	Information Length	0x04
0x02	Message ID	0x05 (COT)
0x03	Parameter Count	0x01
0x04	Parameter 1 ID	0x10 (Continuity Indicators)
0x05	Parameter 1 Data Length	0x00
0x06–0x09	Not Used	
0x0A	Continuity Recheck	0x00=Enables unlimited repeated continuity checks 0x01= Limits the repeated continuity checks to n+2 times where n is the value of config byte 0x0B. (default)
0x0B	Number of times to recheck	2 (default)
0x14	Total data length of release message	0x06
0x15	Release message type	0x0C
0x16	Parameter count	0x01
0x17	Parameter Type	0x12

Byte	Description	Value
0x18	Parameter Data Length	0x02
0x19	Data[0]	0x80
0x1A	Data[1]	0xA9

ISUP CRO PPL Event Indications (ANSI)

Event ID	Event
0x01	Subsequent Continuity Recheck attempt successful
0x02	T5 expiry in Wait for Release Complete State
0x03	1st Continuity Recheck attempt failure
0x04	Continuity Recheck attempt failure
0x05	1st Continuity Recheck attempt successful

ISUP CRO PPL Timers (ANSI)

Timer ID	Timer	Default Value (10ms)
0x01	T1	1500
0x05	T5	6000
0x19	T25	1000
0x0A	T26	18000

ISUP SSC (0x0085)

Purpose This section includes ITU PPL Config Bytes and Timer information for the PPL Component ISUP SSC.

ISUP SSC Configuration Bytes (ITU)

Byte	Description	Value
0x01	Note Used	
0x02	SGM indication to host	0 Concatenate Message with SGM and report to host. 1 (default) Send Separate SGMindication (event: 0x38, comp: 0x0F)
0x03	Not used	
0x04	Outgoing Segmentation length limit	1 (default) (MSB) This makes it 0x104 which is equal to 260 octets.
0x05		4 (default) (LSB)

ISUP SSC PPL Timers (ITU)

Timer ID	Timer	Default Value (10ms)
0x01	T34	2000

ISUP ACC (0x00A8)

Purpose This section includes PPL Config Bytes and Timer information for the PPL Component ISUP ACC (Automatic Congestion Control).

ISUP ACC Configuration Bytes

Byte	Description	Value
0x01	Enable/Disable ACC Remote Algorithm	0x00=Disable 0x01=Enable (default)
0x02	Remote Node ACC or non-ACC	0x00=non-ACC node 0x01=ACC node (default)
0x03	Enable/Disable Network/Link Congestion Control	0x00=Disable 0x01=Enable (default)
0x14	Throttle Timer Ticks in Seconds	0x01=one second
0x1E	Average Timer Ticks in Seconds	0x04=four seconds
0x1F	Alpha Value of Average Timer normalized to 128	64 dec (.5) 0x40
0x20	Beta Value of Average Timer normalized to 128	64 dec (.5) 0x40
0x21	X Value of Average Timer normalized to 128	102 dec (.2) 0x66
0x22	Y Value of Average Timer normalized to 128	25 (.8) 0x19

Byte	Description	Value
0x23	Threshold (Leaky Bucket Size) of Remote Node for Congestion Level 1	10 decimal 0x0A
0x24	Threshold (Leaky Bucket Size) of Remote Node for Congestion Level 2	5 decimal 0x05
0x25	Increment Factor for Remote Node	10 decimal 0x0A
0x28	Network Congestion Control (NCC) Multiplicative Decrement Constant Beta_0 normalized to 128	128 decimal 0x80
0x29	NCC Multiplicative Decrement Constant Beta_1 normalized to 128	122 decimal 0x7A
0x2A	NCC Multiplicative Decrement Constant Beta_2 normalized to 128	110 decimal 0x6E
0x2B	NCC Multiplicative Decrement Constant Beta_3 normalized to 128	90 decimal 0x5A
0x2C	NCC Multiplicative Decrement Constant Beta_4 normalized to 128	40 decimal 0x28
0x2D	NCC Additive Increase Constant Alpha	8 decimal 0x08

Byte	Description	Value
0x32	NCC Handling of TFC	0x00=Multi levels with priorities(ANSI, CHINA, JT) 0x01= Single Level without priorities(ITU) 0x02=Multi levels without priorities(BT)

PPL Timers

Timer ID	Timer	Default Value
0x01	Short Timer (T_short)-T29	60 (600 ms) 0x3C
0x02	Long Timer (T_long) - T30	1200 (12 seconds) 0x4B0
0x03	Tick Timer (T_tick) which drives Throttle Timer and Average Timer	100 (1 second) 0x64
0x04	NCC Timer (T15+T16)	500 (5 seconds) 0x1F4

ISUP PPL Information for Other PPL Components

Purpose

The tables below show the ITU and ANSI PPL Event Indications and PPL Timers supported for ISUP components not previously described.The following events are only informational indications, and require no specific host response.

ISUP PPL Event Indications for Other PPL Components (ITU)

PPL Component	Event ID	Event
BLS (0x0016)	0x01	First Expiration of T13
	0x02	First Expiration of T15
CRS (0x0018)	0x01	First Expiration of T17
CRCR (0x0044)	0x01	REL received with check indicator set to ‘3’
	0x02	T27 timer expiration
	0x03	Second failure of COT
CGRS (0x001B)	0x01	First Expiration of T23
	0x02	UCIC Stop request received
CGRR (0x001C)	0x01	Wait for CPC Reset TO, remote reset failed.
MGBS (0x0045)	0x01	First Expiration of T19
	0x02	First Expiration of T21
	0x03	Stop event received in Wait for CGBA state
	0x04	Stop event received in Wait for CGUA state
	0x0A	CGBA Status Bit Error Handling function. (Outgoing bit set to 0. Incoming bit set to 1.)
	0x0B	CGBA Status Bit Error Handling function. (Outgoing bit set to 1. Incoming bit set to 0.)

PPL Component	Event ID	Event
	0x0C	CGUA Status Bit Error Handling function. (Outgoing bit set to 0. Incoming bit set to 1.)
	0x0D	CGUA Status Bit Error Handling function. (Outgoing bit is set to 1. Incoming bit is set to 0.)
HGBS (0x0047)	0x01	First Expiration of T19
	0x02	First Expiration of T21
	0x03	Stop event received in Wait for CGBA state
	0x04	Stop event received in Wait for CGUA state
	0x0A	CGBA Status Bit Error Handling function. (Outgoing bit set to 0. Incoming bit set to 1.)
	0x0B	CGBA Status Bit Error Handling function. (Outgoing bit set to 1. Incoming bit set to 0.)
	0x0C	CGUA Status Bit Error Handling function. (Outgoing bit set to 0. Incoming bit set to 1.)
	0x0D	CGUA Status Bit Error Handling function. (Outgoing bit is set to 1. Incoming bit is set to 0.)

ISUP PPL Event Indications for Other PPL Components (ANSI)

The table below shows the ANSI PPL Event Indications supported for ISUP components not previously described. The following events are only informational indications, and require no specific host response.

PPL Component	Event ID	Event
BLS (0x0016)	0x01	Expiration of T13
	0x02	Expiration of T15
CGRR (0x001C)	0x01	Failure to receive L3p Group Reset response

PPL Component	Event ID	Event
CGRS (0x001B)	0x01	First Expiration of T23
	0x02	UCIC Stop request received
CRS (0x0018)	0x01	First Expiration of T17
GBUS (0x001D)	0x01	Expiration of T19
	0x02	Expiration of T21
	0x03	CGBA status bit mismatch
	0x04	CGUA status bit mismatch
	0x05	Incorrect “circuit group supervision message type indicator” parameter in received circuit group (un)blocking acknowledgment.

ISUP PPL Timers for Other PPL Components (ITU)

The table below shows the ITU PPL Timer values for ISUP PPL components not previously described.

PPL Component	PPL Timer ID	Timer	Default Value (10ms)
BLS (0x0016)	0x01	T12	1500
	0x02	T13	36000
	0x03	T14	1500
	0x04	T15	36000
	0x05	White Book	6000
CRS (0x0018)	0x01	T16	1500
	0x02	T17	6000
CRCR (0x0044)	0x01	T27	24000
	0x02	T36	1200
CGRS (0x001B)	0x01	T22	1500
	0x02	T23	30000
CGRR (0x001C)	0x01	Wait for CPC Response Timer	1500

PPL Component	PPL Timer ID	Timer	Default Value (10ms)
HGBS (0x0047)	0x01	T18	1500
	0x02	T19	30000
	0x03	T20	1500
	0x04	T21	30000
	0x05	White Book	6000
MGBS (0x0045)	0x01	T18	1500
	0x0 2	T19	30000
	0x03	T20	1500
	0x04	T21	30000
	0x05	White Book	6000

ISUP PPL Timers for Other PPL Components (ANSI)

The table below shows lists the ANSI PPL Timer values for ISUP PPL components not previously described.

PPL Component	Timer ID	Timer	Default Value (10ms)
BLS (0x0016)	0x01	T12	1500
	0x02	T13	6000
	0x03	T14	1500
	0x04	T15	2000
CRI (0x0015)	0x01	TCCR,r	2000
	0x02	T27	18000
	0x03	T34	1500
CRS (0x0018)	0x01	T16	1500
	0x02	T17	6000
CGRS (0x001B)	0x01	T22	1500
	0x02	T23	6000

PPL Component	Timer ID	Timer	Default Value (10ms)
CGRR (0x001C)	0x01	Wait for CPC Response Timer	1500
GBUS (0x001D)	0x01	T18	1500
	0x02	T19	6000
	0x03	T20	1500
	0x04	T21	6000

L3P TUP (0x0011)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component L3P TUP.

L3P TUP Configuration Bytes

Byte	Description	Value
0x01	Answer Type	0x00=ANC (default) 0x01=ANN 0x02=ANU
0x02	CCL Reception Processing Options	0x00=Auto-reply w/ CBK (default) 0x01=Send PPL Event (CCL) to host
0x03	Outgoing Call Release Option	0x00=CLF (default) 0x01=CCL
0x04	Host-initiated OOS	0x00=BLK/UBL (default) 0x01=RSC
0x05	CBK Reception Processing Options	0x00=Auto-reply with CLF (default) 0x01=PPL Event (CBK) to host
0x06	RFS Format	0x00=TUP Fields (default) 0x01=BCD Encoded Digits
0x07	Send ACM PPL Event to host	0x00=Do Not Send (default) 0x01=Send
0x08	Send Answer PPL Event to host	0x00=Do Not Send (default) 0x01=Send
0x09	Outgoing Congestion Control	0x00 - Disable 0x01 - Enable (default)

Byte	Description	Value
0x0A	Enable/Disable PPL Event Indication sending to host. 1=Send PPL Event Indication to host. 0=Do not send PPL Event Indication to host. If this byte is set to “1”, all the block/unblock acknowledgments and reset complete indications (RLG) will be sent to the host.	0x00 **NOTE:** Config Byte 0x0A is used only for China TUP customized PPLs.
0x0B	Send Auto ACM 0=Wait for Host initiated ACM 1=Generate ACM using default parameters	0x00
0x0C-0x0E	Unused	
0x0F	Field ID of Called Party Address	0x03
0x10	Field ID of Calling Line ID	0x08
0x11	Address Indicator for Calling Line ID	00x0
0x12-0x13	Unused	
Address Complete Message (ACM)		
0x14	Number of Fields	0x01
0x15	ACM Message Indication Fields	0x0C
0x16	Field Length	0x01
0x17	Field Data	0x00
0x18-0x1D	Unused	
General Request Message (GRQ)		

Byte	Description	Value
0x1E	Number of Fields	0x01
0x1F	GRQ Message Indication Fields	0x0D
0x20	Field Length	0x01
0x21	Field Data	0x00
0x22-0x27	Unused	
Automatic Congestion Control Message (ACC)		
0x28	Number of Fields	0x01
0x29	ACC Message Indication Fields	0x13
0x2A	Field Length	0x01
0x2B	When set will cause a REL to be sent to the network prior to transitioning into the In Service Protocol.	0x00 (default) 0x01 - Send REL to Network
0x2C-0x31	Unused	
Extended Unsuccessful Backward Setup Information Message (EUM)		
0x32	Number of Fields	0x02
0x33	EUM Octet Indication Fields	0x14
0x34	Field Length	0x01
0x35	Field Data	0x01
0x36	SPC	0x15
0x37	Field Length	0x02
0x38	Field Data	0x00
0x39	Field Data	0x00
0x3A-0x3B	Unused	
Initial Address Message (IAM)		
0x3C	Number of Fields	0x02
0x3D	Calling Party Category	0x01

Byte	Description	Value
0x3E	Field Length	0x01
0x3F	Field Data	0x0A
0x40	Field ID	0x02 - Call Setup Message Indicators
0x41	Field Length	0x02
0x42	Field Data	0x00
0x43	Field Data	0x00
0x44-0x4F	Unused	
Initial Address Message with Initial Information (IAI)		
0x50	Number of Fields	0x02
0x51	Calling Party Category	0x01
0x52	Field Length	0x01
0x53	Field Data	0x0A
0x54	Field ID	0x02 - Call Setup Message Indicators
0x55	Field Length	0x02
0x56	Field Data	0x00
0x57	Field Data	0x00

L3P TUP PPL Event Indications

Event ID	Event
0x28	Received GSM
0x29	Received SAO/SAM
0x2A	Received FOT
0x2B	Received RAN
0x2C	Received CCL
0x2D	Received GRQ
0x2E	Received ACM
0x2F	Received Answer (ANC, ANN, ANU)
0x30	Received Call Unsuccessful after ACM (Congestion)
0x31	Received CBK
0x32	Remotely Maintenance Blocked
0x33	Remotely Hardware Blocked
0x34	Remotely Software Blocked
0x35	Remotely Maintenance Unblocked
0x36	Remotely Hardware Unblocked
0x37	Remotely Software Unblocked
0x38	Local Maintenance Block Request Successfully Completed
0x39	Local Maintenance Block Request Unsuccessful
0x3A	Local Maintenance Unblock Request Successfully Completed
0x3B	Local Maintenance Unblock Request Unsuccessful
0x3C	Local Software Block Request Successfully Completed
0x3D	Local Software Block Request Unsuccessful
0x3E	Local Software Unblock Request Successfully Completed
0x3F	Local Software Unblock Request Unsuccessful
0x40	Local Maintenance Blocking Cleared.

Event ID	Event
0x41	Local Software Blocking Cleared
0x43	ACC Received
0x44	Layer 5 Group Reset/Reset Request Successful
0x45	Layer 5 Group Reset/Reset Request Unsuccessful
0x64	TUP Protocol Violation
0x0191	Setup Indication (containing ICB data IAM/IAI)
0x0192	Channel Release with ICB data
0x0193	Reset Indication
0x0194	Loopback Indication
0x0195	Loopback Remove Indication
0x0196	Circuit INS Indication
0x0197	Clear Forward Indication
0x0198	Invalid ICB data message discarded
0x0199	Circuit OOS Indication
0x019A	Call unsuccessful indication with ICB data (CFL)
0x019B	RLG Indication
0x019C	Group Reset Indication
0x019D	Reset Complete Indication
0x019E	Clearback Receive Indication, CLF sent
0x019F	Virtual CIC channel status after Matrix Controller switchover
Outgoing Call Failure reasons are listed below as Event Indications 0x013B-0x0149. These occur before receiving a backward message (equivalent to outseize NACK).	
0x013B	Outgoing Call rejected, channel blocked
0x0147	Outgoing Call rejected, invalid channel state or an error indication from TUP
0x0148	Outgoing Call rejected, dual seizure

Event ID	Event
0x0149	Outgoing Call rejected, local circuit reset due to an internal error

L3P TUP PPL Event Requests

Event ID	Event
0x01	Send ACM
0x02	Send CFL
0x03	Send GRQ
0x04	Send Call Unsuccessful
0x05	Send Answer
0x06	Send CBK
0x07	Send RAN
0x08	Send SAO/SAM
0x09	Send GSM
0x0A	Send FOT
0x0B	Send CCL
0x0C	Send ACC
0x0D	Locally Maintenance Block Request (Individual)
0x0E	Locally Maintenance Unblock Request (Individual)
0x0F	Locally Maintenance Block Request (Group)
0x10	Locally Maintenance Unblock Request (Group)
0x11	Locally Software Block Request (Group)
0x12	Locally Software Unblock Request (Group)
0x13	Circuit Reset Request
0x14	Circuit Group Reset Request
0x15	Hardware Block Request (Group)

Event ID	Event
0x16	Hardware Unblock Request (Group)
0x3C	Send IAM/IAI
0x3D	Send Release
0x3E	Send Loopback ACK for incoming COT
0x3F	Span OOS status
0x40	Span INS status

L3P TUP PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	L4 Loopback ACK Wait	1000
0x02	L4 Clear Request Wait	3000
0x03	TUP Reset Complete Wait	6000
0x04	M/SW Block Response Wait	12000
0x05	HW Block for OOS Response Wait	12000

TUP CPC (0x0052)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP CPC.

TUP CPC Configuration Bytes

Byte	Description	Value
0x0A	RLG Field Count	0x00
0x14	CFL Field Count	0x00
0x1E	CLF Field Count	0x00
0x28	Incoming Congestion Control	0x00=Disable 0x02=Enable (default)
0x2D	Congestion Type (as defined by TUP section 3.7.2)	1=Switching Equipment Congestion 2=Circuit Group Congestion 3=National Network

TUP CPC PPL Event Indications

Event ID	Event
0x01	COT Received
0x02	CCF Received
0x03	Failure to receive CLF in response to Call Unsuccessful CFL
0x04	T5 Expiration
0x05	T7 Expiration
0x06	Continuity in Incoming IAM

TUP CPC PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T1	1500
0x02	T4	1500
0x03	T5	6000
0x04	T3	1500
0x05	T2	2000
0x06	T6	1500
0x07	T7	6000

TUP SPRC (0x0053)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP SPRC.

TUP SPRC Configuration Bytes

Byte	Description	Value
0x01	SIO Subservice Field	0x08
0x02	SIO Service Indicator	0x04
0x0A	Pause/Resume Option	0x00
0x0B–0x27	Unused	
Congestion Control		
0x28	Number of probes to determine Average Delay	10
0x29	Probe Delay Threshold for Incoming Calls, 100 ms units	100 ms
0x2A	Average Delay Threshold for Incoming Calls, 100 ms units	50 ms
0x2B	Congestion Cleared Threshold for Incoming Calls, 100 ms units	10 ms
0x2C	Number of probes to determine congestion cleared=Incoming Calls	10
0x2D	Probe Delay Threshold for Outgoing Calls, 100 ms units	100 ms
0x2E	Average Delay Threshold for Outgoing Calls, 100 ms units	50 ms
0x2F	Congestion Cleared Threshold for Outgoing Calls, 100 ms units	10 ms
0x30	Number of probes to determine congestion cleared=Outgoing Calls	10
0x31–0xC8	Unused	

TUP SPRC PPL Event Indications

Event ID	Event
0x01	Unknown TUP message type received
0x02	Bad message format in received message
0x03	Message received to UCIC

TUP SPRC PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	Pause Timer Tick	400

TUP BLR (0x0054)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP BLR.

TUP BLR Configuration Bytes

Byte	Description	Value
0x0A	BLA Field Count	0x00
0x1E	UBA Field Count	0x00

TUP BLR PPL Event Indications

Event ID	Event
0x01	T11 Expiration
0x02	T25 Expiration

TUP BLR PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T11	30000
0x02	T25	30000

TUP BLS (0x0055)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP BLS.

TUP BLS Configuration Bytes

Byte	Description	Value
0x0A	BLO Field Count	0x00
0x1E	UBL Field Count	0x00

TUP BLS PPL Event Indications

Event ID	Event
0x01	T13 Expiration
0x02	T11 Expiration
0x03	T25 Expiration
0x04	T16 Expiration

TUP BLS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T11	30000
0x02	T12	1500
0x03	T13	6000
0x04	T14	6000
0x05	T15	1500
0x06	T16	6000
0x07	T17	6000
0x08	T25	30000

TUP CRI (0x0057)

Purpose This section includes PPL Config Bytes and all PPL Events for the PPL Component TUP CRI.

TUP CRI Configuration Bytes

Byte	Description	Value
0x0A	RLG Field Count	0x00

TUP CRI PPL Event Indications

Event ID	Event
0x01	CLF with first time indicator set
0x02	CCF with first time indicator on
0x03	CCF received

TUP CRS (0x0058)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP CRS.

TUP CRS Configuration Bytes

Byte	Description	Value
0x0A	RSC Field Count	0x00
0x14	RLG Field Count	0x00

TUP CRS PPL Event Indications

Event ID	Event	
0x01	First T19 expiration	

TUP CRS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T18	1500
0x02	T19	6000

TUP MBUS (0x005B)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP MBUS.

TUP MBUS Configuration Bytes

Byte	Description	Value
0x0A	MGB Field Count	0x00
0x1E	MGU Field Count	0x00

TUP MBUS PPL Event Indications

Event ID	Event
0x01	First T27
0x02	First T29

TUP MBUS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T26	1500
0x02	T27	6000
0x03	T28	1500
0x04	T29	6000

TUP MBUR (0x005C)

Purpose This section includes Timer information for the PPL Component TUP MBUR.

TUP MBUR PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T23	500
0x02	T24	500

TUP CGRR (0x0059)

Purpose This section includes PPL Config Bytes and Timer information for the PPL Component TUP CGRR.

TUP CGRR Configuration Bytes

Byte	Description	Value
0x0A	GRA Field Count without range and status	0x00

TUP CGRR PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T20	500
0x02	L3P Protection Timer	500

TUP CGRS (0x005A)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP CGRS.

TUP CGRS Configuration Bytes

Byte	Description	Value
0x0A	GRS Field Count	0x00

TUP CGRS PPL Event Indication

Event ID	Event
0x01	First T22

TUP CGRS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T21	1500
0x012	T22	6000

ISUP CGRS (0x001B)

Purpose This section includes PPL Config Bytes for the PPL Component ISUP CGRS.

ISUP CGRS Configuration Bytes

Byte	Description	Value
0x01	This config byte determines the number of retries of GRS that will be sent to the network when an unequipped CIC (UCIC) is received in a response. The circuit group reset (GRS) is retried n times or until a circuit group reset acknowledgment GRA is received (whichever is earlier).	0x00 (default) The valid range is 0-255.

TUP HBUS (0x005D)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP HBUS.

TUP HBUS Configuration Bytes

Byte	Description	Value
0x0A	HGB Field Count	0x00
0x1E	HGU Field Count	0x00

TUP HBUS PPL Event Indications

Event ID	Event
0x01	First T33
0x02	First T35

TUP HBUS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T32	1500
0x02	T33	6000
0x03	T34	1500
0x04	T35	6000

TUP HBUR (0x005E)

Purpose This section includes Timer information for the PPL Component TUP HBUR.

TUP HBUR PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T30	500
0x02	T31	500

TUP SBUS (0x005F)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component TUP SBUS.

TUP SBUS Configuration Bytes

Byte	Description	Value
0x0A	MGB Field Count	0x00
0x1E	MGU Field Count	0x00

TUP SBUS PPL Event Indications

Event ID	Event
0x01	First T39
0x02	First T41

TUP SBUS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T38	1500
0x02	T39	6000
0x03	T40	1500
0x04	T41	6000

TUP SBUR (0x0060)

Purpose This section includes Timer information for the PPL Component TUP SBUR.

TUP SBUR PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T36	500
0x02	T37	500

L3P SSUTR2 (0x0011)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component L3P SSUTR2, a variant of L3P TUP.

L3P SSUTR2 Configuration Bytes

Block 1

Byte	Description	Value
0x01	Host initiated OOS	0=BLO/UBL (default) 1=RZC
0x02	RAU reception processing options	0=Auto reply w FIU (default) 1=PPL Event Indication RAU to Host (Host should send FIU)
0x03	RFS Format	0=SSUTR2 field format (default) 1=BCD encoded digits
0x04	Send ACF PPL Event Indication to host	0=Do not send (default) 1=Send event to host
0x05	Send Answer PPL Event Indication to host	0=Do not send (default) 1=Send event to host
0x06	Congestion control check	0=Congestion control disabled 1=Congestion control enabled (default)
0x07	Specialized Circuit	0=Mixed circuit (incoming/ outgoing) (default) 1=Incoming calls only
0x0F	Field ID of called party address	0x03
0x10	Field ID of calling party address	0x08

Byte	Description	Value
0x11	Address indicator for calling line ID	0
0x14	No. of Fields in DEG	1
0x15	DEG message indicator field ID	0x0D
0x16	Field len	0x01
0x17	Field Data [0]	0x01
0x1E	MIF No. of fields	0x02
0x1F	Calling party category field ID	0x01
0x20	Field len	0x01
0x21	Field Data [0]	0x0A
0x22	MIF message indicator field ID	0x02
0x23	Field len	0x02
0x24	Field Data [0]	0x00
0x25	Field Data [1]	0x00

Block 3

Byte	Description	Value
0x01	ACF No. of fields	0x03
0x02	ACF message indicator field ID	0x0C
0x03	Field len	0x01
0x04	Field Data [0]	0x00
0x05	ACF nature of access field ID	0x18
0x06	Field len	0x01
0x07	Field Data [0]	0x01
0x08	ACF circuit info field ID	0x19
0x09	Field len	0x01
0x0A	Field Data [0]	0x01

L3P SSUTR2 PPL Event Indications

Event ID	Event
0x28	Received IFG
0x29	Received MSA/MSS
0x2B	Received NRP
0x2D	Received DEG
0x2E	Received ACF
0x2F	Received RIU
0x30	Received call unsuccessful
0x31	Received RAU
0x32	Outgoing (Remote) maintenance block indication
0x35	Outgoing (Remote) maintenance unblock indication
0x38	Incoming (Local) maintenance block request successfully completed
0x39	Incoming (Local) maintenance block request unsuccessful
0x3A	Incoming (Local) maintenance unblock request successfully completed

Event ID	Event
0x3B	Incoming (Local) maintenance unblock request unsuccessful
0x40	Incoming (Local) maintenance blocking clear indication
0x44	L5 reset/ group reset successful
0x45	L5 reset/ group reset unsuccessful
0x64	TUP protocol violation
0x65	Resources not available for continuity check
0x0259	Received end to end message MUU/MCE
0x025A	Received CHT
0x025B	Received ITX
0x025C	Received TAX
0x025D	Received TXA
0x025E	System error block request successfully completed
0x025F	System error block request unsuccessful
0x0260	System error unblock request successfully completed
0x0261	System error unblock request unsuccessful
0x0262	Outgoing block unsuccessful (Already blocked)
0x0263	Outgoing unblock unsuccessful (Not blocked)
0x0264	BLOF operation unsuccessful
0x0265	BLOF clear unsuccessful

L3P SSUTR2 PPL Event Requests

Event ID	Event
0x01	Send ACF (Alerting)
0x02	Send ECH (Call failure message)
0x03	Send DEG (General Setup Request)
0x04	Send Call Unsuccessful (Call unsuccessful signal)
0x05	Send RIU (Answer)
0x06	Send RAU (Clear Back)
0x07	Send NRP (Reanswer)
0x08	Send MSS/MSA (Subsequent address)
0x09	Send IFG (General setup information)
0x0D	Send BLOA (manual incoming blocking)
0x0E	Send BLOA clear (unblock incoming)
0x13	Circuit reset request
0x14	Circuit group reset request
0x64	Send CCD (Manual Outgoing Continuity Recheck)
0x65	Stop CCD (Stop Outgoing Continuity Recheck)
0x66	Send End to End Message MUU/MCE
0x67	Local BLOF
0x68	Local BLOD (manual outgoing blocking)
0x69	Local BLOD (manual outgoing blocking) clear
0x6A	Local BLOS (Blocking due to internal system fault)
0x6B	Local BLOS (Blocking due to internal system fault) clear
0x6C	Send TAX
0x6D	Send CHT
0x6E	Send ITX
0x6F	Local BLOF clear

L3P SSUTR2 PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	L4 Loopback ACK wait	1000
0x02	L4 Clear req wait	3000
0x03	SSUTR2 reset complete wait	6000
0x04	Maintenance block response wait	12000
0x05	OOS block response wait	12000

SSUTR2 CPC (0x0052)

Purpose This section includes PPL Config Bytes, all PPL Event Indications, and Timer information for the PPL Component SSUTR2 CPC, a variant of TUP CPC.

SSUTR2 CPC Configuration Bytes

Byte	Description	Value
0x0A	LIG (Release Guard Field Count)	0
0x14	ECH (Call Failure Signal Field Count)	0
0x19	Specialized Circuit (Location of this byte must not be changed)	0=Mixed circuit (incoming/outgoing) (default) 1=Outgoing call only
0x1A	Congestion Control Enabled/Disabled	0=Disabled 1=Enabled (default)
0x1B	Congestion Signal Type	1=EEC 2=EFC 3=ERN
0x1E	FIU (Clear Forward Field Count)	0

SSUTR2 CPC PPL Event Indications

Event ID	Event
0x01	Incoming Continuity Check successful
0x02	Incoming Continuity Check failure
0x03	Failure to receive FIU in response to call unsuccessful ECH
0x04	Stop ECH repeat after timeout
0x05	Stop FIU repeat after timeout
0x06	Continuity in incoming MIF
0x07	Outgoing continuity success
0x08	Outgoing continuity failure

SSUTR2 CPC PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	Wait for subsequent address message Tsam	1500
0x02	Release guard timer1 Tlig	1000
0x03	Release guard timer2 T(FIU repeat)	6000
0x04	Wait for 1st backward signal timer Tacf	2400
0x05	Non receipt of reply signal Triu	24000
0x06	Non receipt of FIU after sending clearback Tfiu	3000
0x07	Non hang-up of the caller after receipt of a hang-up signal of the called subscriber	500
0x08	Non receipt of continuity test message Tcont	1500
0x09	Non receipt of the end signal after transmission of a call setup failure signal	1000s
0x0A	ECH repetition stop T(ECH, repeat)	6000

SSUTR2 SPRC (0x0053)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component SSUTR2 SPRC, a variant of TUP SPRC.

SSUTR2 SPRC Configuration Bytes

Byte	Description	Value
0x01	SIO subservice field	8
0x02	SIO service indicator	4
0x0A	Pause/Resume option	0
0x28	Number of samples, n, in moving average	100
0x29	Instantaneous Onset Threshold for Incoming Calls, 100 ms units	100 ms
0x2A	Moving Average Onset Threshold for Incoming Calls, 100 ms units	50 ms
0x2B	Abatement Threshold for Incoming Calls, 100 ms units	10 ms
0x2C	Number of samples, x, received before clearing congestion condition for Incoming Calls	10
0x2D	Instantaneous Onset Threshold for Outgoing Calls, 100 ms units	100 ms
0x2E	Moving Average Onset Threshold for Outgoing Calls, 100 ms units	50 ms
0x2F	Abatement Threshold for Outgoing Calls, 100 ms units	10 ms
0x30	Number of samples, x, received before clearing congestion condition for Outgoing Calls	10

SSUTR2 SPRC PPL Event Indications

Event ID	Event
0x01	Incoming Continuity Check successful
0x02	Incoming Continuity Check failure
0x03	Failure to receive FIU in response to call unsuccessful ECH
0x04	Stop ECH repeat after timeout
0x05	Stop FIU repeat after timeout
0x06	Continuity in incoming MIF
0x07	Outgoing continuity success
0x08	Outgoing continuity failure

SSUTR2 SPRC PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	Pause Timer Tick	400
0x02	Q probe timer	100

SSUTR2 BLR (0x0054)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component SSUTR2 BLR, a variant of TUP BLR.

SSUTR2 BLR Configuration Bytes

Byte	Description	Value
0x0A	BLA Field Count	0
0x1E	UBA Field Count	0

SSUTR2 BLR PPL Event Indications

Event ID	Event
0x01	T11 Expiration
0x02	T25 Expiration

SSUTR2 BLR PPL Timers

Timer ID	Timer	Value
0x01	TUP T11	30000

SSUTR2 BLS (0x0055)

Purpose This section includes PPL Config Bytes and Timer information for the PPL Component SSUTR2 BLS, a variant of TUP BLS.

SSUTR2 BLS Configuration Bytes

Byte	Description	Value
0x0A	BLO Field Count	0
0x1E	UBL Field Count	0

SSUTR2 BLS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	T13 Expiration	100
0x02	T11 Expiration	200
0x03	T25 Expiration	300
0x04	T16 Expiration	400
0x14	Maintenance block request for outgoing specialized circuit not allowed	2000

SSUTR2 CRI (0x0057)

Purpose This section includes all PPL events and Timer information for the PPL Component SSUTR2 CRI, a variant of TUP CRI.

SSUTR2 CRI PPL Event Indications

Event ID	Event
0x01	FIU in 1st recheck attempt
0x02	CCN in 1st recheck attempt

SSUTR2 CRI PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	Wait for FIU	1000

SSUTR2 CRS (0x0058)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component SSUTR2 CRS, a variant of TUP CRS.

SSUTR2 CRS Configuration Bytes

Byte	Description	Value
0x0A	RZC Field Count	0
0x1E	LIG Field Count	0

SSUTR2 CRS PPL Event Indications

Event ID	Event
0x01	1st T19 Expiration

SSUTR2 CRS PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	TUP T18	1500
0x02	TUP T19	6000

SUTR2 CRO (0x007C)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component SSUTR2 CRO.

SSUTR2 CRO Configuration Bytes

Byte	Description	Value
0x0A	Recheck retry after failure (0-disabled, 1-Enabled)	1
0x0B	Number of retries	2
0x0F	FIU field count	0

SSUTR2 CRO PPL Event Indications

Event ID	Event
0x01	Retries exhausted after T8 expiry
0x02	Dual seizure, continuity recheck stopped
0x03	FIU repeat timer expiry, reset initiated
0x04	Error: Recheck attempt on non-idle circuit
0x05	LIG received indication
0x06	Continuity Recheck attempt successful after 1st retry

SSUTR2 CRO PPL Timers

Timer ID	Timer	Default Value (10ms)
0x01	TUP T9	1000
0x02	TUP T8	200
0x03	TUP T10	18000
0x04	Wait for LIG	1000
0x05	FIU repeat	6000

L3P BT IUP (0x0011)

L3P BT IUP Configuration Bytes

Block 1

The L3P BT IUP configuration bytes below are included in Block 1.

Byte	Description of Block 1 Byte	Value
0x01	Generate PPL Event Indication for IAM Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x02	Generate PPL Event Indication for IFAM Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x03	Generate PPL Event Indication for SAM Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x04	Generate PPL Event Indication for FAM Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x05	Generate PPL Event Indication for Send “N” Digits Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x06	Generate PPL Event Indication for Send All Digits Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x07	Generate PPL Event Indication for Send Additional Setup Information Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x08	Generate PPL Event Indication for Address Complete Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x09	Generate PPL Event Indication for Congestion Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x0A	Generate PPL Event Indication for Terminal Congestion Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x0B	Generate PPL Event Indication for Connection Not Admitted Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x0C	Generate PPL Event Indication for Repeat Attempt Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x0D	Generate PPL Event Indication for Subscriber Engaged Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x0E	Generate PPL Event Indication for Subscriber Out of Order Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x0F	Unused	
0x10	Unused	
0x11	Generate PPL Event Indications for Answer Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x12	Generate PPL Event Indication for Clear Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x13	Generate PPL Event Indication for Re-Answer Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x14	Generate PPL Event Indication for Release Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x15	Unused	
0x16	Unused	
0x17	Unused	
0x18	Unused	
0x19	Generate PPL Event Indications for Circuit Free Bit Number Message 0 Generate State 2-7 Unused	Note 2 0 (default)
0x1A	Generate PPL Event Indication for Blocking Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x1B	Generate PPL Event Indication for Unblocking Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x1C	Generate PPL Event Indication for Block Ack Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x1D	Generate PPL Event Indication for Unblock Ack Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x1E	Generate PPL Event Indication for Overload Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x1F	Unused	
0x20	Unused	
0x21	Unused	
0x22	Unused	
0x23	Unused	
0x24	Unused	
0x25	Generate PPL Event Indication for Confusion Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x26	Generate PPL Event Indication for ISDN Composite Service Info Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x27	Unused	
0x28	Unused	
0x29	Generate PPL Event Indications for Additional Call Information Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)
0x2A	Unused	
0x2B	Unused	
0x2C	Generate PPL Event Indication for Swap Bit Number Message 0 Generate State 2-7 Unused	See Note 2. 0 (default)

Byte	Description of Block 1 Byte	Value
0x2D	For the AUTO (IAM-SND) protocol: The number of accumulated digits received must be greater than or equal to this value to prevent another SND message from being sent. For the AUTO (IAM-SAD) protocol: The number of accumulated digits received must be greater than or equal to this value to prevent another SAD message from being sent. See config block 1, byte 0x2E for setting of the AUTO (IAM-SND) or AUTO (IAM-SAD) modes. NOTE: If more than 30 digits are received the channel will be purged.	0 = Default
0x2E	IAM Operational Mode. Auto(IAM-SND)/Host/AUTO(IAM-SAD) control of messaging on reception of an IAM. In Auto IAM-SND Mode the PPL code will automatically attempt to retrieve the Desires number of Digits based upon the setting in the above config byte using the SND protocol. In Host mode the Host must request any desired additional digits, and ACI messages, if required. It must deal with the networks responses appropriately. In the AUTO IAM-SAD mode the PPL code will automatically attempt to retrieve at least the desired number of digits based upon the setting in the above config byte or timer 0x16 expiring or the reception of a FAM using the SAD protocol.	0 = Default AUTO IAM-SND 1 = Host CTRL 2 = AUTO IAM-SAM

Byte	Description of Block 1 Byte	Value
0x2F	IFAM Operational Mode. This controls the PPL processing by selection of one of two actions. L3p takes these actions after the IFAM is received and a Setup message is sent to L4. In the Basic processing mode, the PPL waits for L4 to issue an Alerting message, or any other Host desired message wrapped in a PPL Event Request. In the Auto mode, the PPL will issue an ACI/ASUI message to L3 and await it's response. Once received it will send a PPL Event Indication of ACI/ASUI to the Host. ACI is sent whenever the received Interworking bit in the IAM/IFAM message, is zero and ASUI when the bit is a one. The ACI/ ASUI responses will be generated by the PPL code whenever a valid request is received regardless of this config byte's setting. Auto response to ACI request messages. PNI processing. When enabled, and the PNI bit is set will cause up to 2 Auto ACI requests to be issued. As defined in config block 1 bytes 0x60 and 0x61.	Bit 0 0 = Default, Auto ACI/ASUI 1 = Basic Bit 1 0 = Default, Auto respond to ACI request. 1 = Do not respond automatically to ACI request. Allow Host to respond. Bit 2 0 = Disabled 1 = Enabled

Byte	Description of Block 1 Byte	Value
0x30	L3p IUP Setup Report Format. Bit Number Message 0 ICB encoding to use whenever L3pIUP is required to issue a Setup msg to L4. 1-7 Unused	0, Report using actual IUP ICBs = Default. 1, Report only the called address (optionally calling address) using Stage N Address Data ICB in BCD.
0x31	Network Configuration Bit Number Description 0 Unused 1 Unidirectional/Both way. 2 Exchange Type	0 = Default, Both way 1= Unidirectional 0 = Terminating 1 = Intermediate
0x32	Out Of Service to Inservice transition notification. 0 = Send UBL (Unblock) to network 1 = Send nothing	0 = Default
0x33	Control Reporting of Call Connected upon receiving ANS. 0 = Suppress Message 1 = Issue PPL Event Indication message	1 = Default

Byte	Description of Block 1 Byte	Value
0x34	The RFS with data will not be reported to the HOST until after the results of the ACI/SASUI is collected. The Host will be presented with a single IUP ICB containing the original IAM/ IFAM message and the received ACI/SASUI response. When this byte is set, config block 1, byte 0x2F is ignored, but it's functionality is activated to get the ACI/ASUI responses	0 = Default Disable late RFS reporting. 1 = Enable RFS reporting after auto ACI request/response.
0x35	Outgoing Congestion Control	0= Enabled- Default. 1 = Disabled.
0x36	For the AUTO (IAM-SND) or AUTO (IAM-SAD) protocol: The Minimum number of accumulated digits when a FAM is received. If this minimum is not satisfied the call will be denied via a CAN. Use config block 1, byte 0x2D to set the minimum number of IAM/SAM digits to receive.	0 = Default
0x37	Number of digits to match. If these digits match the digits in the accumulated digit buffer then use the value in config block 1, byte 0x3D in stead of config byte 0x2D to limit the number of SNDs/SADs to issue. And use the value in config block 1, byte 0x3E in stead of config byte 0x36 to set the minimum number of digits required when a FAM is received.	0 = Default
0x38	First two digits to match	0 = Default
0x39	Second two digits to match	0 = Default
0x3A	Third two digits to match	0 = Default
0x3B	Fourth two digits to match	0 = Default
0x3C	Fifth two digits to match	0 = Default
0x3D	Alternate minimum number of digits that must be received to prevent another SND or SAD to be issued. Used instead of config block 1, byte 0x2D.	0 = Default
0x3E	Alternate minimum number of digits that must be received to successfully allow a FAM to complete the address information. Used in stead of config block 1, byte 0x36.	0 = Default

Byte	Description of Block 1 Byte	Value
0x3F	If PPL Event Request of ANM is sent to L3P to automatically put, L4 in the answer state.	0 - Do not put L4 in answered state automatically (default) 1 - Put L4 in answered state automatically
0x40-0x5F	Unused	
0x60	When PNI processing is enabled, The Auto ACI IRC value to be used when the PNI bit is clear, or the second Auto ACI IRC value to be used when the PNI is set. When PNI processing is disabled this contains the Auto ACI IRC value.	0 = Don't issue Request. 1 = Default
0x61	When PNI processing is enabled, The first Auto ACI IRC value to be used when the PNI bit is set.	0 = Don't issue Request. 0 = Default
0x62	Preserve Block states when DPC becomes inaccessible.	0 = Preserve 1 = Destroy
0x63	Preserve (Maintain) Answered Calls when DPC becomes inaccessible.	0 = Preserve 1 = Destroy
0x64	Beginning of **IAM** Message. Number of TLVs that follow	 4
0x65	**H0 H1** element **TLV ID**	1
0x66	Length of TLV data to follow	2
0x67	First byte (i.e. H0) of H0 H1	0
0x68	Second byte (i.e. H1) of H0 H1	0
0x69	IAM/IFAM Message Ind. TLV ID	2
0x6A	Length of TLV data to follow	5
0x6B-0x6F	Data for this TLV	
0x70	Called Address TLV ID	3

Byte	Description of Block 1 Byte	Value
0x71	Length of TLV data to follow	Variable 2 to 0x0B, 0x0B = Default
0x72-0x7C	Data for this TLV	
0x7D	Line Identity TLV	4
0x7E	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x7F-0x87	Data for this TLV	
0x88	Beginning of **IFAM** Message. Number of TLVs that follow	4
0x89	**H0 H1** element **TLV ID**	1
0x8A	Length of TLV data to follow	2
0x8B-0x8C	Data for this TLV	
0x8D	IAM/IFAM Message Ind. TLV ID	2
0x8E	Length of TLV data to follow	5
0x8F-0x93	Data for this TLV	
0x94	Called Address TLV ID	3
0x95	Length of TLV data to follow	variable 2 to 0x0B, 0x0B = Default
0x96-0xA0	Data for this TLV	
0xA1	Line Identity TLV ID	4
0xA2	Length of TLV data to follow	variable 2 to 9, 9 = Default
0xA3-0xAB	Data for this TLV	

Byte	Description of Block 1 Byte	Value
0xAC	Beginning of **SAM** Message. Number of TLVs that follow	2
0xAD	**H0 H1** element **TLV ID**	1
0xAE	Length of TLV data to follow	2
0xAF-0xB0	Data for this TLV	
0xB1	Reserved Data Byte TLV ID	0x18
0xB2	Length of TLV data	1
0xB3	Reserved byte	0
0xB4	SAM/FAM Called Address TLV ID	0x19
0xB5	Length of TLV data to follow	variable 2 to 0x0A, 0x0A = Default
0xB6-0xBF	Data for this TLV	
0xC0	Beginning of **FAM** Message. Number of TLVs that follow	2
0xC1	**H0 H1** element **TLV ID**	1
0xC2	Length of TLV data to follow	2
0xC3-0xC4	Data for this TLV	
0xC5	Reserved Data Byte TLV ID	0x18
0xC6	Length of TLV data	1
0xC7	Reserved byte	0
0xC8	SAM/FAM Called Address TLV ID	0x19

Block 2

The L3P BT IUP configuration bytes below are included in Block 2.

Byte	Description of Block 2 Byte	Value
0x01	Length of TLV data to follow	variable 2 to 0x0A, 0x0A = Default
0x02- 0x0B	Data for this TLV	
0x0C	Beginning of **ASUI Type 1** Message. Number of TLVs that follow	3
0x0D	**H0 H1** element **TLV ID**	1
0x0E	Length of TLV data to follow	2
0x0F- 0x10	Data for this TLV	
0x11	ASUI Message Information Contained Code TLV ID	5
0x12	Length of TLV data to follow	1
0x13	Data for this TLV	0
0x14	Line Identity TLV ID	4
0x15	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x16- 0x1E	Data for this TLV	
0x1F	Beginning of **ASUI Type 2** Message. Number of TLVs that follow	4
0x20	**H0 H1** element **TLV ID**	1
0x21	Length of TLV data to follow	2
0x22- 0x23	Data for this TLV	
0x24	ASUI Message Information Contained Code TLV ID	5

Byte	Description of Block 2 Byte	Value
0x25	Length of TLV data to follow	1
0x26	Data for this TLV	0
0x27	Reserved Data Byte TLV ID	0x18
0x28	Length of TLV data to follow	1
0x29	Data for this TLV	0
0x2A	Partial Calling Line Identity TLV ID	6
0x2B	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x2C-0x34	Data for this TLV	
0x35	Beginning of **SND** Message. Number of TLVs that follow	2
0x36	**H0 H1** element **TLV ID**	1
0x37	Length of TLV data to follow	2
0x38-0x39	Data for this TLV	
0x3A	Number of Digits Requested TLV ID	7
0x3B	Length of TLV data to follow	1
0x3C	Data for this TLV	0
0x3D	Beginning of **SAD** Message. Number of TLVs that follow	2
0x3E	**H0 H1** element **TLV ID**	1
0x3F	Length of TLV data to follow	2
0x40-0x41	Data for this TLV	
0x42	Reserved Data Byte TLV ID	0x18
0x43	Length of TLV data to follow	1
0x44	Reserved byte	0

Byte	Description of Block 2 Byte	Value
0x45	Beginning of **SASUI** Message. Number of TLVs that follow	2
0x46	**H0 H1** element **TLV ID**	1
0x47	Length of TLV data to follow	2
0x48-0x49	Data for this TLV	
0x4A	SAUSI Information Indicator TLV ID	8
0x4B	Length of TLV data to follow	1
0x4C	Data for this TLV	
0x4D	Beginning of **ACM** Message. Number of TLVs that follow	2
0x4E	**H0 H1** element **TLV ID**	1
0x4F	Length of TLV data to follow	2
0x50-0x51	Data for this TLV	
0x52	Address Complete Message Indicator TLV ID	9
0x53	Length of TLV data to follow	3
0x54-0x56	Data for this TLV	
0x57	Beginning of **CNG** Message. Number of TLVs that follow	1
0x58	**H0 H1** element **TLV ID**	1
0x59	Length of TLV data to follow	2
0x5A-0x5B	Data for this TLV	
0x5C	Beginning of **TCM** Message. Number of TLVs that follow	1
0x5D	**H0 H1** element **TLV ID**	1
0x5E	Length of TLV data to follow	2

Byte	Description of Block 2 Byte	Value
0x5F-0x60	Data for this TLV	
0x61	Beginning of **CAN** Message. Number of TLVs that follow	3
0x62	**H0 H1** element **TLV ID**	1
0x63	Length of TLV data to follow	2
0x64-0x65	Data for this TLV	
0x66	CAN Reason TLV ID	0x0A
0x67	Length of TLV data to follow	1
0x68	Data for this TLV	
0x69	CAN Diagnostics TLV ID	0x0B
0x6A	Length of TLV data to follow	1
0x6B	Data for this TLV	
0x6C	Beginning of **ANS** Message. Number of TLVs that follow	2
0x6D	**H0 H1** element **TLV ID**	1
0x6E	Length of TLV data to follow	2
0x6F-0x70	Data for this TLV	
0x71	Type of Answer TLV ID	0x0B
0x72	Length of TLV data to follow	1
0x73	Data for this TLV	
0x74	Beginning of **CLR** Message. Number of TLVs that follow	1
0x75	**H0 H1** element **TLV ID**	1
0x76	Length of TLV data to follow	2
0x77-0x78	Data for this TLV	

Byte	Description of Block 2 Byte	Value
0x79	Beginning of **REL** Message. Number of TLVs that follow	2
0x7A	**H0 H1** element **TLV ID**	1
0x7B	Length of TLV data to follow	2
0x7C-0x7D	Data for this TLV	
0x7E	Release Reason TLV ID	0x0D
0x7F	Length of TLV data to follow	1
0x80	Data for this TLV	0x2F
0x81	Beginning of **CCF** Message. Number of TLVs that follow	1
0x82	**H0 H1** element **TLV ID**	1
0x83	Length of TLV data to follow	2
0x84-0x86	Data for this TLV	
0x86	Beginning of **BLO** Message. Number of TLVs that follow	1
0x87	**H0 H1** element **TLV ID**	1
0x88	Length of TLV data to follow	2
0x89-0x8A	Data for this TLV	
0x8B	Beginning of **UBL** Message. Number of TLVs that follow	1
0x8C	**H0 H1** element **TLV ID**	1
0x8D	Length of TLV data to follow	2
0x8E-0x8F	Data for this TLV	
0x90	Beginning of **BLA** Message. Number of TLVs that follow	1
0x91	**H0 H1** element **TLV ID**	1

Byte	Description of Block 2 Byte	Value
0x92	Length of TLV data to follow	2
0x93-0x94	Data for this TLV	
0x95	Beginning of **UBA** Message. Number of TLVs that follow	1
0x96	**H0 H1** element **TLV ID**	1
0x97	Length of TLV data to follow	2
0x98-0x99	Data for this TLV	
0x9A	Beginning of **CFN** Message. Number of TLVs that follow	2
0x9B	**H0 H1** element **TLV ID**	1
0x9C	Length of TLV data to follow	2
0x9D-0x9E	Data for this TLV	
0x9F	Confusion Message Qualifier TLV ID	0x0E
0xA0	Length of TLV data to follow	1
0xA1	Data for this TLV	
0xA2	Beginning of **ACI Type 1** Message. Number of TLVs that follow	5
0xA3	**H0 H1** element **TLV ID**	1
0xA4	Length of TLV data to follow	2
0xA5-0xA6	Data for this TLV	
0xA7	ACI Information Contained and Information Requested TLV ID	0x0F
0xA8	Length of TLV data to follow	2
0xA9-0xAA	Data for this TLV	
0xAB	ACI Type 1 Reserved Bytes TLV ID	0x10
0xAC	Length of TLV data to follow	2

Byte	Description of Block 2 Byte	Value
0xAD-0xAE	Data for this TLV	
0xAF	ACI Type 1 Message Indicators TLV ID	0x11
0xB0	Length of TLV data to follow	1
0xB1	Data for this TLV	
0xB2	Line Identity TLV ID	4
0xB3	Length of TLV data to follow	variable 2 to 9, 9 = Default
0xB4-0xBC	Data for this TLV	
0xBD	Beginning of **ACI Type 2** Message. Number of TLVs that follow	3
0xBE	**H0 H1** element **TLV ID**	1
0xBF	Length of TLV data to follow	2
0xC0-0xC1	Data for this TLV	
0xC2	ACI Information Contained and Information Requested TLV ID	0x0F
0xC3	Length of TLV data to follow	2
0xC4-0xC5	Data for this TLV	
0xC6	Partial Calling Line Identity TLV ID	6
0xC7	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0xC8	Data for this TLV	

Block 3

The L3P BT IUP configuration bytes below are included in Block 3.

	Description for Block 3 Byte	**Value**
1-8	Data for last byte of this TLV	
9	Beginning of **ACI Type 3** Message. Number of TLVs that follow	5
0x0A	**H0 H1** element **TLV ID**	1
0x0B	Length of TLV data to follow	2
0x0C-0x0D	Data for this TLV	
0x0E	ACI Information Contained and Information Requested TLV ID	0x0F
0x0F	Length of TLV data to follow	2
0x10-0x11	Data for this TLV	
0x12	Calling/Called Subscriber's Basic Service Marks TLV ID	0x12
0x13	Length of TLV data to follow	2
0x14-0x15	Data for this TLV	
0x16	ACI Type 3 Message Indicators TLV ID	0x13
0x17	Length of TLV data to follow	1
0x18	Data for this TLV	
0x19	Line Identity TLV ID	4
0x1A	Length of TLV data to follow	variable 2 to 9, 9 = Default
0x1B-0x23	Data for this TLV	
0x24	Beginning of **ACI Type 4** Message. Number of TLVs that follow	4
0x25	**H0 H1** element **TLV ID**	1
0x26	Length of TLV data to follow	2
0x27-0x28	Data for this TLV	

	Description for Block 3 Byte	**Value**
0x29	ACI Information Contained and Information Requested TLV ID	0x0F
0x2A	Length of TLV data to follow	2
0x2B-0x2C	Data for this TLV	
0x2D	Calling/Called Subscriber's Basic Service Marks TLV ID	0x12
0x2E	Length of TLV data to follow	2
0x2F-0x30	Data for this TLV	
0x31	ACI Type 4 Message Indicators TLV ID	0x14
0x32	Length of TLV data to follow	1
0x33	Data for this TLV	
0x34	Beginning of **ACI Type 5** Message. Number of TLVs that follow	3
0x35	**H0 H1** element **TLV ID**	1
0x36	Length of TLV data to follow	2
0x37-0x38	Data for this TLV	
0x39	ACI Information Contained and Information Requested TLV ID	0x0F
0x3A	Length of TLV data to follow	2
0x3B-0x3C	Data for this TLV	
0x3D	Calling Subscriber's Originating Facility Marks TLV ID	0x15
0x3E	Length of TLV data to follow	2
0x3F-0x40	Data for this TLV	
0x41	Beginning of **ACI Type 6** Message. Number of TLVs that follow	3
0x42	**H0 H1** element **TLV ID**	1
0x43	Length of TLV data to follow	2

	Description for Block 3 Byte	**Value**
0x44-0x45	Data for this TLV	
0x46	ACI Information Contained and Information Requested TLV ID	0x0F
0x47	Length of TLV data to follow	2
0x48-0x49	Data for this TLV	
0x4A	Called Subscriber's Terminating Facility Marks TLV ID	0x16
0x4B	Length of TLV data to follow	2
0x4C-0x4D	Data for this TLV	
0x4E	Beginning of **ACI Type 7** Message. Number of TLVs that follow	2
0x4F	**H0 H1** element **TLV ID**	1
0x50	Length of TLV data to follow	2
0x51-0x52	Data for this TLV	
0x53	ACI Information Contained and Information Requested TLV ID	0x0F
0x54	Length of TLV data to follow	2
0x55-0x56	Data for this TLV	
0x57	Beginning of **OLM** Message. Number of TLVs that follow	1
0x58	**H0 H1** element **TLV ID**	1
0x59	Length of TLV data to follow	2
0x5A-0x5B	Data for this TLV	
0x5C	Beginning of **SOO** Message. Number of TLVs to follow	1
0x5D	**H0 H1** element **TLV ID**	1

	Description for Block 3 Byte	**Value**
0x5E	Length of TLV data to follow	2
0x5F-0x60	Data for this TLV	
0x61	Beginning of **SEM** Message. Number of TLVs to follow	1
0x62	**H0 H1** element **TLV ID**	1
0x63	Length of TLV data to follow	2
0x64-0x65	Data for this TLV	
0x66	Beginning of **SIM Type 1** Message Number of TLVs to follow	2
0x67	**H0 H1** element **TLV ID**	1
0x68	Length of TLV data to follow	2
0x69-0x6A	Data for this TLV	
0x6B	Information Contained and Information Requested	0x0F
0x6C	Length of TLV data to follow	2
0x6D-0x6E	Data for this TLV	
0x6F	Beginning of **SIM TYPE 2** Message Number of TLVs to follow	6
0x70	**H0 H1** element **TLV ID**	1
0x71	Length of TLV data to follow	2
0x72-0x73	Data for this TLV	
0x74	Information Contained and Information Requested	0x0F
0x75	Length of TLV data to follow	2
0x76-0x77	Data for this TLV	
0x78	Facility Indicator Code	0x1B
0x79	Length of TLV data to follow	2

	Description for Block 3 Byte	**Value**
0x7A-0x7B	Data for this TLV	
0x7C	Service Indicator Code	0x1D
0x7D	Length of TLV data to follow	2
0x7E-0x7F	Data for this TLV	
0x80	Reserved Data Byte	0x18
0x81	Length of TLV data to follow	1
0x82	Data for this TLV	
0x83	Line Identity (LI)	4
0x84	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x85-0x8D	Data for this TLV	
0x8E	Beginning of **SIM TYPE 3** Message Number of TLVs to follow	7
0x8F	**H0 H1** element **TLV ID**	1
0x90	Length of TLV data to follow	2
0x91-0x92	Data for this TLV	
0x93	Information Contained and Information Requested	0x0F
0x94	Length of TLV data to follow	2
0x95-0x96	Data for this TLV	
0x97	Facility Indicator Code	0x1B
0x98	Length of TLV data to follow	2
0x99-0x9A	Data for this TLV	
0x9B	Service Indicator Code	0x1D
0x9C	Length of TLV data to follow	2

	Description for Block 3 Byte	**Value**
0x9D-0x9E	Data for this TLV	
0x9F	Closed User Group Interlock Code	0x1C
0xA0	Length of TLV to follow	4
0xA1-0xA4	Data for this TLV	
0xA5	Reserved Data Byte	0x18
0xA6	Length of TLV data to follow	1
0xA7	Data for this TLV	
0xA8	Line Identity (LI)	4
0xA9	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0xAA-0xB2	Data for this TLV	
0xB3	Beginning of **SIM TYPE 4** Message Number of TLVs to follow	4
0xB4	**H0 H1** element **TLV ID**	1
0xB5	Length of TLV data to follow	2
0xB6-0xB7	Data for this TLV	
0xB8	Information Contained and Information Requested	0x0F
0xB9	Length of TLV data to follow	2
0xBA-0xBB	Data for this TLV	
0xBC	Reserved Data Byte	0x18
0xBD	Length of TLV data to follow	1
0xBE	Data for this TLV	
0xBF	Line Identity (LI)	4
0xC0	Length of TLV data to follow	Variable 2 to 9, 9 = Default

	Description for Block 3 Byte	Value
0xC1-0xC8	Data for this TLV	

Block 4

The L3P BT IUP configuration bytes below are included in Block 4.

Byte	Description of Block 4 Byte	Value
0x01	Yet more Data for this TLV	
0x02	Beginning of **SIM TYPE 5** Message Number of TLVs to follow	7
0x03	**H0 H1** element **TLV ID**	1
0x04	Length of TLV data to follow	2
0x05-0x06	Data for this TLV	
0x07	Information Contained and Information Requested	0x0F
0x08	Length of TLV data to follow	2
0x09-0x0A	Data for this TLV	
0x0B	Facility Indicator Code	0x1B
0x0C	Length of TLV data to follow	2
0x0D-0x0E	Data for this TLV	
0x0F	Service Indicator Code	0x1D
0x10	Length of TLV data to follow	2
0x11-0x12	Data for this TLV	
0x13	Reserved Data Byte	0x18
0x14	Length of TLV data to follow	1
0x15	Data for this TLV	
0x16	Line Identity (LI)	4

Byte	Description of Block 4 Byte	Value
0x17	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x18- 0x20	Data for this TLV	
0x21	Six Character Network Address Extension	0x1E
0x22	Length of TLV data to follow	Variable 2 to 7 7 = Default
0x23- 0x29	Data for this TLV	
0x2A	Beginning of **SIM TYPE 6** Message Number of TLVs to follow	8
0x2B	**H0 H1** element **TLV ID**	1
0x2C	Length of TLV data to follow	2
0x2D- 0x2E	Data for this TLV	
0x2F	Information Contained and Information Requested	0x0F
0x30	Length of TLV data to follow	2
0x31- 0x32	Data for this TLV	
0x33	Facility Indicator Code	0x1B
0x34	Length of TLV data to follow	2
0x35- 0x36	Data for this TLV	
0x37	Service Indicator Code	0x1D
0x38	Length of TLV data to follow	2
0x39- 0x3A	Data for this TLV	
0x3B	Closed User Group Interlock Code	0x1C
0x3C	Length of TLV data to follow	4
0x3D- 0x40	Data for this TLV	

Byte	Description of Block 4 Byte	Value
0x41	Reserved Data Byte	0x18
0x42	Length of TLV data to follow	1
0x43	Data for this TLV	
0x44	Line Identity (LI)	4
0x45	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x46-0x4E	Data for this TLV	
0x4F	Six Character Network Address Extension	0x1E
0x50	Length of TLV data to follow	Variable 2 to 7 7 = Default
0x51-0x57	Data for this TLV	
0x58	Beginning of **SIM TYPE 7** Message Number of TLVs to follow	3
0x59	**H0 H1** element **TLV ID**	1
0x5A	Length of TLV data to follow	2
0x5B-0x5C	Data for this TLV	
0x5D	Information Contained and Information Requested	0x0F
0x5E	Length of TLV data to follow	2
0x5F-0x60	Data for this TLV	
0x61	Facility Indicator Code	0x1B
0x62	Length of TLV data to follow	2
0x63-0x64	Data for this TLV	
0x65	Beginning of **SIM TYPE 8** Message Number of TLVs to follow	5
0x66	**H0 H1** element **TLV ID**	1

Byte	Description of Block 4 Byte	Value
0x67	Length of TLV data to follow	2
0x68-0x69	Data for this TLV	
0x6A	Information Contained and Information Requested	0x0F
0x6B	Length of TLV data to follow	2
0x6C-0x6D	Data for this TLV	
0x6E	Facility Indicator Code	0x1B
0x6F	Length of TLV data to follow	2
0x70-0x71	Data for this TLV	
0x72	Reserved Data Byte	0x18
0x73	Length of TLV data to follow	1
0x74	Data for this TLV	
0x75	Line Identity (LI)	4
0x76	Length of TLV data to follow	Variable 2 to 9, 9 = Default
0x77-0x7F	Data for this TLV	
0x80	Beginning of **Send Service** Message Number of TLVs to follow	1
0x81	**H0 H1** element **TLV ID**	1
0x82	Length of TLV data to follow	2
0x83-0x84	Data for this TLV	
0x85	Beginning of **Service** Message Number of TLVs to follow	2
0x86	**H0 H1** element **TLV ID**	1
0x87	Length of TLV data to follow	2
0x88-0x89	Data for this TLV	

Byte	Description of Block 4 Byte	Value
0x8A	Service Code TLV ID	0x1F
0x8B	Length of TLV data to follow	1
0x8C	Data for this TLV	
0x8D	Beginning of **Suspend (SUS)** Message Number of TLVs to follow	2
0x8E	**H0 H1** element **TLV ID**	1
0x8F	Length of TLV data to follow	2
0x90-0x91	Data for this TLV	
0x92	Reserved Data Byte	0x18
0x93	Length of TLV data to follow	1
0x94	Data for this TLV	
0x95	Beginning of **Resume (RES)** Message Number of TLVs to follow	2
0x96	**H0 H1** element **TLV ID**	1
0x97	Length of TLV data to follow	2
0x98-0x99	Data for this TLV	
0x9A	Reserved Data Byte	0x18
0x9B	Length of TLV data to follow	1
0x9C	Data for this TLV	
0x9D	Beginning of **Repeat Attempt (RAM)** Message Number of TLVs to follow	1
0x9E	**H0 H1** element **TLV ID**	1
0x9F	Length of TLV data to follow	2
0xA0-0xA1	Data for this TLV	
0xA2	Beginning of **Re-answer (RAN)** Message Number of TLVs to follow	1

Byte	Description of Block 4 Byte	Value
0xA3	**H0 H1** element **TLV ID**	1
0xA4	Length of TLV data to follow	2
0xA5-0xA6	Data for this TLV	
0xA7	Beginning of **SWAP** Message Number of TLVs to follow	3
0xA8	**H0 H1** element **TLV ID**	1
0xA9	Length of TLV data to follow	2
0xAA-0xAB	Data for this TLV	
0xAC	Service Indicator Code	0x1D
0xAD	Length of TLV data to follow	2
0xAE-0xAF	Data for this TLV	
0xB0	Reserved Data Byte	0x18
0xB1	Length of TLV data to follow	1
0xB2	Data for this TLV	
0xB3	Beginning of **Operator Override** Message Number of TLVs to follow	1
0xB4	**H0 H1** element **TLV ID**	1
0xB5	Length of TLV data to follow	2
0xB6-0xB7	Data for this TLV	
0xB8	Beginning of **Coin and Fee Checking** Message Number of TLVs to follow	1
0xB9	**H0 H1** element **TLV ID**	1
0xBA	Length of TLV data to follow	2
0xBB-0xBC	Data for this TLV	

Byte	Description of Block 4 Byte	Value
0xBD	Beginning of **Extend Call** Message Number of TLVs to follow	1
0xBE	**H0 H1** element **TLV ID**	1
0xBF	Length of TLV data to follow	2
0xC0-0xC1	Data for this TLV	
0xC2	Beginning of **Howler** Message Number of TLVs to follow	1
0xC3	**H0 H1** element **TLV ID**	1
0xC4	Length of TLV data to follow	2
0xC5-0xC6	Data for this TLV	
0xC7-0xC8	Unused	

Block 5

The L3P BT IUP configuration bytes below are included in Block 5.

Byte	Description of Block 5 Byte	Value
0x01-0x61	Unused	
0x62-0x63	Pointer to beginning of HLR TLVs	4,0xC2
0x64-0x65	Pointer to beginning of ECM TLVs	4,0xBD
0x66-0x67	Pointer to beginning of CFC TLVs	4,0xB8
0x68-0x69	Pointer to beginning of OOR TLVs	4,0x63
0x6A-0x6B	Pointer to beginning of SWAP TLVs	4,0xA7

Byte	Description of Block 5 Byte	Value
0x6C-0x6D	Pointer to beginning of RAN TLVs	4,0xA2
0x6E-0x6F	Pointer to beginning of RAM TLVs	4,0x9D
0x70-0x71	Pointer to beginning of RES TLVs	4,0x95
0x72-0x73	Pointer to beginning of SUS TLVs	4,0x8D
0x74-0x75	Pointer to beginning of SER TLVs	4,0x85
0x76-0x77	Pointer to beginning of Send Service TLVs	4,0x80
0x78-0x79	Pointer to beginning of SIM TYPE 8 TLVs	4,0x65
0x7A-0x7B	Pointer to beginning of SIM TYPE 7 TLVs	4,0x58
0x7C-0x7D	Pointer to beginning of SIM TYPE 6 TLVs	4,0x2A
0x7E-0x7F	Pointer to beginning of SIM TYPE 5 TLVs	4,2
0x80-0x81	Pointer to beginning of SIM TYPE 4 TLVs	3,0x63
0x82-0x83	Pointer to beginning of SIM TYPE 3 TLVs	3,0x8E
0x84-0x85	Pointer to beginning of SIM TYPE 2 TLVs	3,0x6F
0x86-0x87	Pointer to beginning of SIM TYPE 1 TLVs	3,0x66
0x88-0x89	Pointer to beginning of SEM TLVs	3,0x61
0x8A-0x8B	Pointer to beginning of SOO TLVs	3,0x5C
0x8C-0x8D	Pointer to beginning of OLM TLVs	3,0x57

Byte	Description of Block 5 Byte	Value
0x8E-0x8F	Pointer to beginning of ACI TYPE 7 TLVs	3,0x4E
0x90-0x91	Pointer to beginning of ACI TYPE 6 TLVs	3,0x41
0x92-0x93	Pointer to beginning of ACI TYPE 5 TLVs	3,0x34
0x94-0x95	Pointer to beginning of ACI TYPE 4 TLVs	3,0x24
0x96-0x97	Pointer to beginning of ACI TYPE 3 TLVs	3,9
0x98-0x99	Pointer to beginning of ACI TYPE 2 TLVs	2,0xBD
0x9A-0x9B	Pointer to beginning of ACI TYPE 1 TLVs	2,0xA2
0x9C-0x9D	Pointer to beginning of CFN TLVs	2,0x9A
0x9E-0x9F	Pointer to beginning of UBA TLVs	2,0x95
0xA0-0xA1	Pointer to beginning of BLA TLVs	2,0x90
0xA2-0xA3	Pointer to beginning of ULB TLVs	2,0x8B
0xA4-0xA5	Pointer to beginning of BLO TLVs	2,0x86
0xA6-0xA7	Pointer to beginning of CCF TLVs	2.0x81
0xA8-0xA9	Pointer to beginning of REL TLVs	2,0x79
0xAA-0xAB	Pointer to beginning of CLR TLVs	2,0x74
0xAC-0xAD	Pointer to beginning of ANS TLVs	2,0x6C
0xAE-0xAF	Pointer to beginning of CAN TLVs	2,0x61

Byte	Description of Block 5 Byte	Value
0xB0-0xB1	Pointer to beginning of TCM TLVs	2,0x5C
0xB2-0xB3	Pointer to beginning of CNG TLVs	2,0x57
0xB4-0xB5	Pointer to beginning of ACM TLVs	2,0x4D
0xB6-0xB7	Pointer to beginning of SASUI TLVs	2,0x45
0xB8-0xB9	Pointer to beginning of SAD TLVs	2,0x3D
0xBA-0xBB	Pointer to beginning of SND TLVs	2,0x35
0xBC-0xBD	Pointer to beginning of ASUI Type 2 TLVs	2,0x1F
0xBE-0xBF	Pointer to beginning of ASUI Type 1 TLVs	2,0x0B
0xC0-0xC1	Pointer to beginning of FAM TLVs	1,0xC0
0xC2-0xC3	Pointer to beginning of SAM TLVs	1,0xAC
0xC4-0xC5	Pointer to beginning of IFAM TLVs	1,0x88
0xC6-0xC7	Pointer to beginning of IAM TLVs 0xC6 contains Block Number 1, 2, 3, 4, or 5 0x67 contains the offset to the byte within the block. Range 1 to 0xC8	1, 0x64

PPL Event Indications

Event ID	Event
0x01F5	ACI Received
0x01F6	SAM Received
0x01F7	FAM Received
0x01F8	Undefined Message Received

Event ID	Event
0x01F9	Call Connected Indication
0x01FA	SND Received
0x01FB	CNG Received
0x01FC	CNA Received
0x01FD	RAM Received
0x01FE	SEM Received
0x01FF	SAD Received
0x0200	CFN Received
0x0201	Error, timer expired sending ACI request
0x0202	Error, timer expired sending SASUI request
0x0203	ASUI Received
0x0204	SASUI Received
0x0205	IAM Received
0x0206	IFAM Received
0x0207	ACM Received
0x0208	TCM Received
0x0209	SOO Received
0x020A	ANS Received
0x020B	CLR Received
0x020C	RAN Received
0x020D	REL Received
0x020E	CCF Received
0x020F	OLM Received
0x0210	BLA Received
0x0211	UBA Received
0x0212	BLO Received
0x0213	UBL Received
0x0214	Circuit already Blocked
0x0215	Circuit Not Blocked

Event ID	Event
0x0216	Remote Block Indication, Circuit already Blocked
0x0217	Remote Unblock Indication, Circuit not Blocked
0x0218	Error Timer Expired, Sending CNA
0x0219	Network ANS received before ACM
0x021A	SSM Received
0x021B	SER Received
0x021C	ECM Received
0x021D	CFC Received
0x021E	OOR Received
0x021F	HLR Received
0x0220	SIM Received
0x0221	SUS Received
0x0222	RES Received
0x0223	SWAP Received

L3P BT IUP to Host--Access Denied Message

Whenever L3P BT IUP sends an Access Denied message to Call Control, two arguments are included: Reason and Call Status.

The table below shows the Reasons that may be sent to the host with Access Denied messages. When placing an outbound call, L3P BT IUP may determine that the call cannot be completed. When this occurs, an Access Denied message is sent to the host with the Reason fields encoded.

Reasons for Access Denied Messages

:

Reason	Description
0x1B	Outseize Failure, No ACK
0x1C	Outseize Failure, Glare detected.
0x25	Invalid Encoding Format

Reason	Description
0x59	Outseize Failure, Call Blocked.

Call Status Values:

Call Status	Description
0x00	Call Inactive
0x01	Call Active. Call is already established on this channel.

PPL Event Requests

Event ID	Event
0x1E	ACM
0x1F	ANS
0x20	Undefined Message
0x21	ASUI-1
0x22	ASUI-2
0x23	SND
0x24	REL
0x25	FAM
0x26	SAM
0x27	BLO
0x28	UBL
0x29	ACI Type 1
0x2A	ACI Type 2
0x2B	ACI Type 3
0x2C	ACI Type 4
0x2D	ACI Type 5
0x2E	ACI Type 6

Event ID	Event
0x2F	ACI Type 7
0x30	SASUI
0x31	SSM
0x32	SER
0x33	ECM
0x34	CFC
0x35	OOR
0x36	HLR
0x37	SIM
0x38	CLR
0x39	Re-answer RAN
0x3A	SUS
0x3B	RES
0x3C	SWAP
0x3D	Block Upon Release

L3P BT IUP PPL Timers

Timer ID	Timer	Default Value	Default Duration
0x01	TO-1	18000	180 seconds
0x02	TO-2 Non-receipt of answer	12000	120 seconds
0x03	TO-3	30000	300 seconds
0x04	TO-4	12000	120 seconds
0x05	Awaiting answer	6000	60 seconds
0x09	Awaiting appropriate backward message after IFAM/IAM/SAM/FAM or Forward Group 7 message is sent during call setup phase	2000	20 seconds
0x0E	Await receipt of additional Information Message	500	5 seconds
0x10	Prolonged Call Setup	18000	180 seconds

Timer ID	Timer	Default Value	Default Duration
0x11	Non-receipt of SAM/FAM in response to SAD	2000	20 seconds
0x12	Non-receipt of SAM or FAM in response to Send N Digits.		Dynamic based on number of requested digits and Exchange Type.
0x14	Hardware Blocking Timer	12000	120 seconds
0x15	TO-21 Call Clear Timer	30000	300 seconds
0x16	IAM-SAD maximum digit collection timer	400	4 seconds --This value must be less than the value set for 0x11.
0x17	Block Timer	12000	120 seconds

L3 BT IUP CPC (0x0052)

Purpose This section includes PPL Config Bytes, Event Requests and Timer information for the PPL Component L3 BT IUP CPC, a variant of TUP CPC.

L3 BT IUP CPC Configuration Bytes

Byte	Definition	Value
0x01	Network Configuration Bit Number Description	0=Controlled Network (default) 1=Controlling Network
0x02	Confusion Message Qualifier value Whenever CPC sends the Confusion message this byte will be used to setup the message qualifier byte.	0 (default)
0x03	Incoming Congestion Control.	0=Enabled (default) 1=Disabled
0x04	Release Procedures	0 = Both way Procedures 1 = Disabled
0x05	Unidirectional Customer Location (i.e. Outgoing Network Location) Used for Idle Release Procedures	0 = Other End 1 = This End
0x06	Bearer Release Outgoing Initiated, Subsequent to sending the third release processing procedure	0 =Assume Circuit is Blocked 1 = Repeatedly send a Release upon expiry of TO-12 2 = Assume circuit is Idle, ready to accept another call.

L3 BT IUP CPC PPL Event Indications

Event ID	Event
0x01F5	Invalid IFAM message received from the Network

Event ID	Event
0x01F6	Invalid SAM message received from the Network
0x01F7	Invalid FAM message received from the Network
0x01F8	Invalid IAM message received from the Network
0x01F9	Invalid ASUI message received from the Network
0x01FA	Invalid SND message received from the Network
0x01FB	Invalid SAD message received from the Network
0x01FC	Invalid SASUI message received from the Network
0x01FD	Invalid ACM message received from the Network
0x01FE	Invalid CNG message received from the Network
0x01FF	Invalid CNA message received from the Network
0x0200	Invalid RAM message received from the Network
0x0201	Invalid ANS message received from the Network
0x0202	Invalid CLR message received from the Network
0x0203	Invalid REL message received from the Network
0x0204	Invalid CCF message received from the Network
0x0205	Invalid BLO message received from the Network
0x0206	Invalid UBL message received from the Network
0x0207	Invalid BLA message received from the Network
0x0208	Invalid UBA message received from the Network
0x0209	Invalid CFN message received from the Network
0x020A	Invalid ACI message received from the Network
0x020B	Invalid SEM message received from the Network
0x020C	Invalid TLV Length
0x020D	Invalid Number of TLVs
0x020E	Glare Indication Caused by IAM
0x020F	Glare Indication Caused by IFAM
0x0210	Invalid SOO message received from the Network
0x0212	Invalid SER message received from the Network
0x0213	Invalid SIM message received from the Network

Event ID	Event
0x0214	Invalid OOR message received from the Network
0x0215	Invalid HLR message received from the Network
0x0216	Invalid SSM message received from the Network
0x0217	Invalid CFC message received from the Network
0x0218	Invalid ECM message received from the Network
0x0219	Invalid OLM message received from the Network
0x021A	Invalid SUS message received from the Network
0x021B	Invalid RES message received from the Network
0x021C	Invalid SWAP message received from the Network
0x021D	Non-receipt of CFC or BLO has occurred
0x021E	Invalid RAN message received from the Network
0x021F	Invalid TCM message received from the Network

L3 BT IUP CPC PPL Timers

Timer ID	Timer	Default Value (10 ms)
0x08	TO-8	300
0x0A	TO-10	250
0x0C	TO-12	250

BT IUP SPRC (0x0053)

Purpose This section includes PPL Config Bytes, all PPL events, and Timer information for the PPL Component BT IUP SPRC, a variant of TUP SPRC.

BT IUP SPRC Configuration Bytes

Byte	Description	Value
0x28	Number of samples in moving average	100
0x29	Instantaneous Onset Threshold for Incoming Calls. In units of 100ms	1000 (10 ms)
0x2A	Moving Average Onset Threshold for Incoming Calls. In units of 100ms.	500 (10 ms)
0x2B	Abatement Threshold for Incoming calls. In units of 100ms	100 (10 ms)
0x2C	Number of samples to received before clearing Incoming congestion.	10
0x2D	Instantaneous Onset Threshold for Outgoing Calls. In units of 100ms.	1000 (10 ms)
0x2E	Moving Average Onset Threshold for Outgoing Calls. In units of 100ms.	500 (10 ms)
0x2F	Abatement Threshold for Outgoing Calls. In units of 100ms.	100 (10 ms)
0x30	Number of samples to receive before clearing Outgoing congestion.	10

BT IUP SPRC PPL Event Indications to Host

Event ID	Event
0x01F4	Unknown Message Received
0x01F5	Message received on Un-equipped CIC

BT IUP SPRC PPL Timers

Timer ID	Timer	Default Value (10ms)
0x02	Time between sending queue probes	100

MTP3 HMDT (0x002B)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component MTP3 HMDT.

MTP3 HMDT Configuration Bytes

Byte	Description	Value
0x01	Variant	0x00=ANSI 0x01=ITU 0x02=Japan TTC/DDI 0x03=China 0x04 = Japan NTT
0x02	Reserved	
0x03	Reserved	
0x04	Network Indicator (NI)	0x00=International 0x01=Reserved 0x02=National (default) 0x03=Reserved
0x05	Not used	
0x06	ANSI Cluster Fanout	0x00 = Disabled (default for all variants except ANSI) 0x01 = Enabled (default for ANSI variant)
0x07	TRW SNM MSU	0x00 = Disabled (default for all variants except ANSI) 0x01 = Enabled (default for ANSI variant)
0x08 to 0x0B	Reserved	
0x0C	Multiple network congestion states	0x00 = Disabled (default for all variants except ANSI and Japan) 0x01 = Enabled (default for ANSI and Japan variants)

Byte	Description	Value
0x0D	UPU unavailability cause	0x00 = Disabled (default for ANSI variant) 0x01 = Enabled (default for all variants except ANSI)
0x0E to 0x0F	Reserved	
0x10	Turn on/off SS7 MTP3 Message Tracing capability for HMDT	0=Off (default) 1=On
	Raw MSU transmitted and received to and from the adjacent Node	0x00=Disable (default) 0x01=Enable the PPL Event Indication containing the LSSU transmitted to the adjacent Node
0x11	TUP Selector	0x00 = TUP (default) 0x01 = BT-IUP
0x12	Japan A/B plane verification	0x00=Disabled (default) 0x01=Enabled
0x13	Japan M/S/U code treatment	0x00 = M/S/U wildcard enabled (default) 0x01 = M/S/U wildcard disabled
0x14 to 0x16	Reserved	
0x17	SCCP User Part Availability	0x00 = Unavailability 0x01 = Available locally (default) 0x02 = Available remotely 0x03 = Invalid
0x18	TUP User Part Availability	0x00 = Unavailable 0x01 = Available locally (default) 0x02 = Available remotely 0x03 = Invalid

Byte	Description	Value
0x19	ISUP User Part Availability	0x00 = Unavailable 0x01 = Available locally (default) 0x02 = Available remotely 0x03 = Invalid
0x1A	User Part 0x06 Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x1B	User Part 0x07 Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available Remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x1C	MTP Test User Part Availability	0x00 = Unavailable 0x01 = Available locally (default) 0x02 = Available remotely 0x03 = Invalid
0x1D	User Part 0x09 Availability	0x00 = Unavailable (default for ITU variant) 0x01 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x1E	User Part 0x0A Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)

Byte	Description	Value
0x1F	User Part 0x0B Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x20	User Part 0x0C Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x21	User Part 0x0D Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x22	User Part 0x0E Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x23	User Part 0x0F Availability	0x00 = Unavailable (default for ITU variant) 0x02 = Available remotely 0x03 = Invalid (default for ANSI and Japan variant)
0x24 to 0x26	Reserved	

Byte	Description	Value
0x27	SCCP User Part Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x28	TUP User Part Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x29	ISUP User Part Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)

Byte	Description	Value
0x2A	User Part 0x06 Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x2B	User Part 0x07 Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x2C	MTP Test User Part Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)

Byte	Description	Value
0x2D	User Part 0x09 Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x2E	User Part 0x0A Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x2F	User Part 0x0B Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)

Byte	Description	Value
0x30	User Part 0x0C Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x31	User Part 0x0D Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x32	User Part 0x0E Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)

Byte	Description	Value
0x33	User Part 0x0F Multihost Routing	0x00 = force routing to port 0x3142 0x01 = force routing to port 0x3143 0x02 = force routing to port 0x3144 0x03 = force routing to port 0x3145 0x04 = force routing to port 0x3146 0xFF = force routing disabled (default)
0x34	Anonymous TFC Treatment	0x00 = Disabled 0x01 = Enabled (default)
0x35	Ericsson TFC Treatment	0x00 = Disabled (default) 0x01 = Enabled
0x36	SCCP Message OPC Validation	0x00 = Do not transmit to user part (default) 0x01 = Transmit to user part (SCCP)

MTP3 HMDT PPL Event Indications

Event ID	Event
0x00	Remote processor outage. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Timestamp 0x0001 Interval sequence number
0x01	Remote processor restored. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Time stamp 0x0001 Interval sequence number

Event ID	Event
0x02	Local inhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Time stamp 0x0001 Interval sequence number
0x03	Local uninhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Timestamp 0x0001 Interval sequence number
0x05	MSU discarded during discrimination. Although MTP3 HMDT is a single state machine per-SS7 signaling stack, this particular event is reported with the SS7 Destination AIB (0x1C) in order to identify the signaling point for which this report applies. Only the 1st occurrence in the current reporting interval for a particular signaling point is reported by this event. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number
0x06	UPU MSU received. Although MTP3 HMDT is a single state machine per-SS7 signaling stack, this particular event is reported with the SS7 Destination AIB (0x1C) in order to identify the signaling point for which this report applies. Only the 1st occurrence in the current reporting interval for a particular signaling point/user part is reported through this event. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x001e User part identifier
0x10	SS7 MTP Message Tracing Not Last

Event ID	Event
0x0014	TRANSFER IND Used to send MSU data. Parameter IDs 0x0424, 0x0425, 0x0426 (see table below)
0x0020	SS7 MTP Message Tracing Not Last (Indicates that ICB data length diagnostic is smaller or equal to 200 bytes).
0x0021	SS7 MTP Message Tracing Last (Indicates that ICB data length diagnostic is greater than 200 bytes).

PPL Parameters

ICB Type	0x03 (Extended Data)
ICB ID	0x0424
Data Length 2 bytes	0x000D
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8-9]	CIC - Circuit ID Code
Data[10]	SI - Service Indicator
Data[11]	MP - Message Priority
Data[12]	NI - Network Indicator

ICB Type	0x03 (Extended Data)
ICB ID	0x0425
Data Length 2 bytes	0x000C
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8]	SI - Service Indicator
Data[9]	MP - Message Priority
Data[10]	NI - Network Indicator

Data[11]	SLS - Signaling Link Set
ICB Type	0x03 (Extended Data)
ICB ID	0x0426
Data Length 2 bytes	Up to 0x010E
Data[0-1]	Length - Length of MSU Data
Data[4-n]	MSU Data - Length up to 0x010C

MTP3 HMDT PPL Event Requests

Event ID	Event
0x01	Start or restart interval reporting. If restarting the measurement interval, this event will result in reports for the current measurement interval before restarting.
0x02	Stop interval reporting. Immediately stop interval reporting.

MTP3 HMRT (0x002C)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component MTP3 HMRT.

MTP3 HMRT Configuration Bytes (ITU/ANSI)

Byte	Description	Value
0x01	Variant	0x00=ANSI 0x01=ITU 0x02=Japan TTC/DDI 0x03=China 0x04=Japan NTT
0x02	MSU Priority Filtering	0x00 = Disabled (ITU default) 0x01 = Enabled (ANSI and Japan default)
0x03	Not Used	
0x04	Network Indicator (NI)	0x00=International 0x01=Reserved 0x02=National (default) 0x03=Reserved
0x05	Reserved	
0x06	Default SNM/SNT MSU Priority	0x00 (ITU default) 0x03 (ANSI and Japan default)
0x07	RCT MSU Priority	0x00 = Dynamic (ANSI default) 0x01 = Static (non-ANSI) default
0x08	ASP Restart Filtering	0x00 = Disable (ANSI default) 0x01 = Enabled (non-ANSI default)
0x09	SP Restart Filtering	0x00 = Disable (ANSI default) 0x01 = Enabled (non-ANSI default)

Byte	Description	Value
0x0A	Reserved	
0x0B	Clear report counters/duration after reporting.	0x00 disabled (default) 0x01 enabled
0x0C	Multi-host routing control. This byte affects all PPL Event Indications in this document. All previously existing PPL Event Indications are not affected.	0x00 force routing to port 0x3142 0x01 force routing to port 0x3143 0x02 force routing to port 0x3144 0x03 force routing to port 0x3145 0x04 force routing to port 0x3146 0xFF force routing disabled (default)
0x0D	PPL Event Compatibility Mode. This byte affects previously existing PPL Event Indications to which the Q.752 Parameters ICB have been added.	0x00=Q.752 Parameters Enabled (default) 0x01=Q.752 Parameters Disabled
0x0E	Reserved	
0x10	Turn on/off SS7 MTP3 Message Tracing capability for HMRT. 1=On 0=Off	0x00
	Raw MSU transmitted and received to and from the adjacent Node	0x00=Disable (default) 0x01=Enable the PPL Event Indication containing the LSSU transmitted to the adjacent Node
0x11	Reserved	

Byte	Description	Value
0x12	Japan A/B plane insertion Setting this configuration byte to 0x01 enables the HMRT component to operate in a Japan DDI network. It enables the proper insertion of the link set using the A/B plane bit for COA and CBA Message Signaling Units when connected to a STP pair using a common point code. Setting this byte to 0x01 should only be used in the Japan DDI network, but not in the Japan JT and Japan NTT networks. This byte has no effect in non-Japan networks.	0x00=Disabled (default) 0x01=Enabled
0x13	ANSI bit rotation	0x00 = Disabled (default for non-ANSI variants 0x01 = Enabled (default for ANSI variant)

MTP3 HMRT PPL Event Indications

Event ID	Event
0x0001	Signaling stack measurement interval report. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x0013 Count of MSUs transmitted 0x0014 Count of octets transmitted 0x0016 Count of MSUs received 0x0017 Count of octets received 0x0019 Count of MSUs discarded during routing 0x001a Count of MSUs discarded during discrimination 0x001b Count of UPUs transmitted for each user part 0x001c Count of UPUs received for each user part

Event ID	Event
0x0002	Signaling link measurement interval report. Although MTP3 HMRT is a single state machine per-SS7 signalling stack, this particular event is reported with the SS7 Link AIB (0x09) in order to identify the signaling link for which this report applies. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x0002 Duration of the link in-service state 0x0003 Duration of the link unavailable state 0x0004 Duration of the link local inhibit state 0x0005 Duration of the link remote inhibit state 0x0006 Duration of the link failed state 0x0007 Duration of the link remote blocked state 0x0008 Count of link failures 0x0009 Count of link restorations 0x000a Count of link failures due to an abnormal FIBR/BSNR 0x000b Count of link failures due to delayed ACK 0x000c Count of link failures due to error rate 0x000d Count of link failures due to excessive congestion duration 0x000e Count of link failures due to remote LSSU received 0x000f Count of link failures due to link test failures 0x0010 Count of link alignment failures 0x0011 Count of link SUs received in error 0x0012 Count of link NACKs received 0x0013 Count of MSUs transmitted 0x0014 Count of octets transmitted 0x0015 Count of link octets retransmitted 0x0016 Count of MSUs received 0x0017 Count of octets received

Event ID	Event
0x0003	Signaling point measurement interval report. Although MTP3 HMRT is a single state machine per-SS7 signaling stack, this particular event is reported with the SS7 Destination AIB (0x1c) in order to identify the signaling point for which this report applies. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x0013 Count of MSUs transmitted 0x0014 Count of octets transmitted 0x0016 Count of MSUs received 0x0017 Count of octets received 0x0018 Duration of inaccessibility 0x0019 Count of MSUs discarded during routing 0x001a Count of MSUs discarded during discrimination 0x001b Count of UPUs transmitted for each user part 0x001c Count of UPUs received for each user part
0x0004	Interval Reports Complete This event indicates reporting for the reporting interval is complete. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number
0x0005	MSU discarded during routing. Although MTP3 HMRT is a single state machine per-SS7 signalling stack, this particular event is reported with the SS7 Destination AIB (0x1c) in order to identify the signalling point for which this report applies. Only the 1st occurrence in the current reporting interval for a particular signaling point is reported by this event. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number

Event ID	Event
0x0006	UPU MSU transmitted. Although MTP3 HMRT is a single state machine per-SS7 signaling stack, this particular event is reported with the SS7 Destination AIB (0x1c) in order to identify the signaling point for which this report applies. Only the 1st occurrence in the current reporting interval for a particular signaling point/user part is reported by this event. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x001e User part identifier
0x0020	SS7 MTP Message Tracing Not Last (Indicates that ICB data length diagnostic is smaller or equal to 200 bytes).
0x0021	SS7 MTP Message Tracing Last (Indicates that ICB data length diagnostic is greater than 200 bytes).

MTP3 HMRT PPL Event Requests (AIB: Stack/Link 0x08)

Event ID	Event
0x0001	Start or restart interval reporting. If restarting the measurement interval, this event will result in reports for the current measurement interval before restarting.
0x0002	Stop interval reporting. Immediately stop interval reporting.
0x0A	TRANSFER REQ Used to send MSU data Parameter ID - 0x0424, 0x0425, 0x0426

PPL Parameters

ICB Type	0x03 (Extended Data)
ICB ID	0x0424
Data Length 2 bytes	0x000D
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8-9]	CIC - Circuit ID Code
Data[10]	SI - Service Indicator
Data[11]	MP - Message Priority
Data[12]	NI - Network Indicator

ICB Type	0x03 (Extended Data)
ICB ID	0x0425
Data Length 2 bytes	0x000C
Data[0-3]	DPC - Destination Point Code
Data[4-7]	OPC - Originating Point Code
Data[8]	SI - Service Indicator
Data[9]	MP - Message Priority
Data[10]	NI - Network Indicator
Data[11]	SLS - Signaling Link Set

ICB Type	0x03 (Extended Data)
ICB ID	0x0426
Data Length 2 bytes	Up to 0x010E
Data[0-1]	Length - Length of MSU Data

Data[4-n]	MSU Data - Length up to 0x010C

MTP3 HMRT PPL Timers

Timer ID	Timer	Default Value (10 ms)
0x01	Monitoring Interval	30000
0x02	Report Pacing	10

MTP3 LSAC (0x002E)

Purpose This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component MTP3 LSAC.

MTP3 LSAC Configuration Bytes

Byte	Description	Value
0x01	Craft Alerting (not supported by Japanese variants)	0x00=Disable (default for ITU) 0x01=Enable (default for ANSI)
0x02	Activation Link Test (cannot be disabled for Japanese variants)	0x00=Disable 0x01=Enable (default)
0x03	Periodic Link Test (not supported by Japanese variants)	0x00=Disable 0x01=Enable (default)

PPL Event Indications Supported by MTP3 LSAC

Event ID	Event
0x0001	Link Activation Failure
0x0002	Signaling link failure. This is an existing PPL Event Indication to which Q.752 Parameters are added (unless operating under backward compatibility). A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number 0x001d Link failure reason code

Event ID	Event
0x0003	Signaling link test failure. This is an existing PPL Event Indication to which Q.752 Parameters are added (unless operating under backward compatibility). A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number
0x0004	Signaling link restoration. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number

MTP3 LSAC PPL Timers (ANSI)

Timer ID	Spec/Timer	Default Value (10 ms)
0x01	T1.111/T19	6000
0x02	T1.111/T17	100
0x03	T1.111/Periodic Link Test Period	6000

MTP3 SLTC (0x0041)

Purpose

The Signaling Link Test Control (SLTC- 0x41) component is used to support the JT telecom standard for Signaling Route Test Control (SRTC).

JT MTP3 PPL Configuration Bytes

The table below shows the configuration bytes for JT MTP3 SLTC.

Byte	Description	Value
0x01	Variant (JT/NTT/DDI)	0x02 NTT/DDI
0x02	Disable non-adjacent received link verify	0x00

JT MTP3 PPL Events

The tables below show the JT PPL Event Requests and Event Indications for MTP3 SLTC.

Requests

Event ID	Event
0x01	Start Signaling Route Test Control (Request)
0x02	Result of Start Signaling Route Test Control (Indication)

Indications

Event ID	Event
0x01	Result of start SRTC

JT MTP3 PPL Timers

The table below shows the JT PPL Timer values for MTP3 SLTC.

Timer ID	Timer	Default Value (1second)
0x01	Supervision timer for Signal Route Test	10 seconds

MTP3 TLAC (0x003C)

Purpose This section includes all PPL events and Timer information for the PPL Component MTP3 TLAC.

MTP3 TLAC PPL Event Indications

Event ID	Event
0x0000	Remote processor outage. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Timestamp 0x0001 Interval sequence number
0x0001	Remote processor restored. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Timestamp 0x0001 Interval sequence number
0x0002	Local inhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Timestamp 0x0001 Interval sequence number
0x0003	Local uninhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Time stamp 0x0001 Interval sequence number

Event ID	Event
0x0004	Remote inhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Time stamp 0x0001 Interval sequence number
0x0005	Remote uninhibit. A single ICB of type Q.752 Parameters is included with the following parameter: 0x0000 Time stamp 0x0001 Interval sequence number

ITU MTP3 TLAC PPL Timers

Timer ID	Spec/Timer	Default Value (10 ms)
0x01	Q.704/T12	100
0x02	Q.704/T13	100
0x03	Q.704/T14	250
0x04	Q.704/T22	18000
0x05	Q.704/T23	18000

MTP3 TLAC PPL Timers (ANSI)

Timer ID	Timer	Default Value (10 ms)
0x01	T12	100
0x02	T13	100
0x03	T14	250
0x04	T20	9000
0x05	T21	9000

MTP3 TSFC (0x003F)

Purpose This section includes all PPL events and Timer information for the PPL Component MTP3 TSFC.

MTP3 TSFC PPL Event Indications

Event ID	Event
0x0001	Destination inaccessible.Although TSFC is a per-stack state machine, this event utilizes a stack/destination AIB to identify the concerned signaling point. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number
0x0002	Destination accessible. TSFC is a per-stack state machine, this event utilizes a stack/destination AIB to identify the concerned signaling point. A single ICB of type Q.752 Parameters is included with the following parameters: 0x0000 Timestamp 0x0001 Interval sequence number
0x0015	PAUSE IND Parameter ID - 0x0427
0x0016	RESUME IND Parameter ID - 0x0427
0x0017	STATUS IND Parameter ID - 0x0427

PPL Parameters

ICB Type	0x03 (Extended Data)
ICB ID	0x0427
Data Length 2 bytes	0x0006
Data[0-3]	Affected DPC
Data[4]	Cause Value Value 0 - congestion level 0 Value 1 - congestion level 1 Value 2 - congestion level 2 Value 3 - congestion level 3 Value 4 - User Part Unavailable Value 5 - User Part Unequipped Value 6 - User Part Inaccessible Value 7 - Signaling Link Congestion level 0 Value 8 - Signaling Link Congestion level 1 Value 9: Signaling Link Congestion level 2 Value 10: Signaling Link Congestion level 3 Value 11: Signaling Link Congestion level 4
Data[5]	User Information Octet value

ITU/ANSI MTP3 Timers for Various Components

Purpose This section contains default settings for additional ITU and ANSI PPL Timers, shown below by PPL component. The MTP3 timers are accurate to 100 ms.

MTP3 PPL Timers (ITU) .

PPL Component	Timer ID	Spec/Timer	Default Value (10 ms)
RCAT (0x0032)	0x01	Q-704/T16	175
RSRT (0x0033)	0x01	Q-704/T10	4500
TCBC (0x0038)	0x01	Q-704/T3	80
	0x02	Q-704/T4	80
	0x03	Q-704/T5	80
TCOC (0x0039)	0x01	Q-704/T1	80
	0x02	Q-704/T2	140
TCRC (0x003A)	0x01	Q-704/T6	80
TPRC (0x003D)	0x01	Q-704/T20	6000
TRCC (0x003E)	0x01	Q-704/T15	250
TSRC (0x0040)	0x01	Q-704/T19	6800
	0x02	Q-704/T21	6400
SLTC (0x0041)	0x01	Q-704/T1	800

MTP3 PPL Timers (ANSI) This section contains default settings for ANSI PPL Timers, shown below by PPL component.

The MTP3 timers are accurate to 100 ms.

.

PPL Component	Timer ID	Spec/Timer	Default Value (10 ms)
RCAT (0x0032)	0x01	T1.111/T16	175
RSRT (0x0033)	0x01	T1.111/T10	4500
TCBC (0x0038)	0x01	T1.111/T3	80
	0x02	T1.111/T4	80
	0x03	T1.111/T5	80
TCOC (0x0039)	0x01	T1.111/T1	80
	0x02	T1.111/T2	140
TCRC (0x003A)	0x01	T1.111/T6	80
TPRC (0x003D)	0x01	T1.111/T22	6000
	0x02	T1.111/T23	1000
	0x03	T1.111/T26	1200
TRCC (0x003E)	0x01	T1.111/T15	250
TSRC (0x0040)	0x01	T1.111/T25	3000
	0x02	T1.111/T28	1000
	0x03	T1.111/T29	6000
SLTC (0x0041)	0x01	T1.111/T1	800

MTP2 TXC (0x0026)

Purpose This section includes PPL Config Bytes and Timer information for the PPL Component MTP2 TXC.

MTP2 TXC Configuration Bytes (ITU/ANSI)

Byte	Description	Value
0x01	LSSU size	0x01 to 0x02 default = 0x01
0x02	Mode	0x00=Basic (default) 0x01=PCR 0x02=Japan TTC
0x03	RTB size in SUs	0x01–0x7F (127) default is 0x38 for Japan variants default=0x7F for non-Japan variants
0x04-0x05	N2 Parameter Value (RTB byte size limit)	0x0117 –0xFFFF (279–65535) default=0xFFFF
0x06	Additional Interframe Flags	0x00–0xFF (default=0x00)
0x07	T_{FISU} Pacing Ticks (Japan TTC only)	0x01–0xFF (default=0x0C (24 ms))
0x08	T_{LSSU} Pacing Ticks (Japan TTC only)	0x01–0xFF (default=0x0C (24 ms))
0x0A-0x0B	Congestion Level 1 Onset Threshold	3200 octets (default) Default value represents 400 ms of queued MSUs at 64 kbps (457 ms@56kbps) Value must be less than or equal to the level 2 onset threshold. Value must be greater than the level 1 abate threshold.

Byte	Description	Value
0x0C-0x0D	Congestion Level 2 Onset Threshold	4150 octets (default) Default value represents 519 ms of queued MSUs at 64 kbps (593 ms @ 56 kbps) This value must be less than or equal to the level 3 onset threshold. This value must be greater than the level 2 abate threshold.
0x0E-0x0F	Congestion Level 3 Onset Threshold	5100 octets (default) Default value represents 638 ms of queued MSUs at 64 kbps (729 ms @ kbps) This value must be less than or equal to the level 4 onset threshold. This value must be greater than the level 3 abate threshold.
0x10-0x11	Congestion Level 4 Onset Threshold	8000 octets (default) Default value represents 1000 ms of queued MSUs at 64 kbps (1143 ms @ 56 kbps) This value must be greater than the level 4 abate threshold.
0x12-0x13	Congestion Level 1 Abate Threshold	1900 octets (default) Default value represents 238 ms of queued MSUs at 64 kbps (271 ms @ 56 kbps) This value must be less than the level 1 onset threshold.

Byte	Description	Value
0x14-0x15	Congestion Level 2 Abate Threshold	3500 octets (default) Default value represents 438 ms of queued MSUs at 64 kbps (500 ms @ 56 kbps) This value must be greater or equal to the level 1 abate threshold. This value must be less than the level 2 onset threshold.
0x16-0x17	Congestion Level 3 Abate Threshold	4500 octets (default) Default value represents 563 ms of queued MSUs at 64 kbps (643 ms @ 56 kbps) This value must be greater than or equal to the level 2 abate threshold. This value must be less than the level 3 onset threshold.
0x18-0x19	Congestion Level 4 Abate Threshold	5500 octets (default) Default value represents 688 ms of queued MSU at 64 kbps (786 ms @ 56 kbps) This value must be greater than or equal to the level 3 abate threshold. This value must be less than the level 4 onset threshold.

MTP2 TXC PPL Timers (ITU/ ANSI)

Timer ID	Spec/Timer	Value
0x01	Q-703/T6 (ITU) T1.111/T6 (ANSI)	400

Timer ID	Spec/Timer	Value
0x02	Q-703/T7 (ITU) T1.111/T7 (ANSI)	100

MTP2 SUERM (0x0025)

Purpose This section includes PPL Config Bytes for the PPL Component MTP2 SUERM.

MTP2 SUERM Configuration Bytes (ITU/ ANSI)

Byte	Description	Value
0x01, 0x02	'T' Parameter=leaky bucket limit threshold (expressed in number of errored SUs)	0x0001–0xFFFF default=0x0040
0x03-0x04	'D' Parameter=leaky bucket drip rate (expressed in number of error-free SUs)	0x0001–0xFFFF default=0x0100

MTP2 IAC (0x0022)

Purpose This section includes a PPL Config Byte and PPL Event for the PPL Component MTP2 IAC.

Configuration Byte

Byte	Description	Value
0x10	Raw LSSU transmitted to adjacent Node The PPL Event Indications for the LSSU received will not contain the IAC component ID.	0x00=Disable (default) 0x01=Enable the PPL Event Indication containing the LSSU transmitted to the adjacent Node.

PPL Event Indication

Event ID	Event
0x0022	LSSU transmitted to adjacent Node

MTP2 LSC (0x0023)

Purpose This section includes a PPL Config Byte and PPL Events for the PPL Component MTP2 LSC.

Configuration Byte

Byte	Description	Value
0x10	Raw LSSU transmitted to and received from the adjacent Node	0x00=Disable (default) 0x01=Enable the PPL Event Indication containing the LSSU transmitted to and received from the adjacent Node.

PPL Event Indications

Event ID	Event
0x0021	LSSU received from adjacent Node
0x0022	LSSU transmitted to adjacent Node

Other MTP2 PPL Timers

Purpose This section contains default settings for ANSI and ITU MTP2 PPL Timers, shown below by PPL component. The MTP2 timers are accurate to 10 ms.

MTP2 PPL Timers for other components (ITU)

PPL Component	Timer ID	Spec/ Timer	Default Value (10 ms)
CC (0x0021)	0x01	Q-703/T5	10
IAC (0x0022)	0x01	Q-703/T2	1150
	0x02	Q-703/T3	150
	0x03	Q-703/T4N	850
	0x04	Q-703/T4E	60
LSC (0x0023)	0x01	Q-703/T1	4500

MTP2 PPL Timers for other components (ANSI)

PPL Component	Timer ID	Timer	Default Value (10 ms)
CC	0x01	T1.111/T5	10
IAC (0x0022)	0x01	T1.111/T2	1150
	0x02	T1.111/T3	1150
	0x03	T1.111/ T4N	230
	0x04	T1.111/T4E	60
LSC (0x0023)	0x01	T1.111/T1	1300

SCCP SCLC (0x0065)

Purpose

This section includes PPL Config Bytes, all PPL Events, and Timer information for the PPL Component SCCP SCLC.

SCCP SCLC Configuration Bytes

Byte	Description	Value
0x01	SCCP Message Format	0x00- International (default for ITU) 0x01=National (default for ANSI) 0x02=China Spec (same as ITU except the point code is 24 bits
0x02	UDTS Message Priority (ANSI Only)	0x02
0x03	N-Unidata *	0x00 - N-Unidata as UDT (Default) 0x01 - N- Unidata as XUDT
0x16-0x19	Used to store the X value.	150 (default)
0x20-0x1D	Used to store the Y value.	200 (default)

* The SCCP user sends SCCP N_UNITDATA to request sending data to a remote SCCP mode. The SCCP then packs the information in N_UNITDATA into an SCCP message and sends the message. The format of the message could be UDT or XUDT.

The XUDT message can be used with GTT to validate the hop counter to detect possible traffic loops. The maximum hop counter is set to 15. We recommend that if you use GTT and if the network supports XUDT, then you should set this SCLC configuration byte 3 to 0x01 to use XUDT message.

This configuration byte only applies to the messages without segmentation. If segmentation is needed, the SCCP will automatically pack and send the XUDT with the segmentation optional parameter.

SCCP SCLC PPL Event Indications

Event ID	Event
0x0A	Maximum reassembly processes are exceeded.

SCCP SCLC PPL Timers

Timer ID	Timer	Default Value (10 ms)
0x01	SCCP Segmentation reassembly timer: Ty Unit is 100 ms.	15000

SCCP SCRC (0x0066)

Purpose This section includes PPL Config Bytes for the PPL Component SCCP SCRC.

SCCP SCRC Configuration Bytes

Byte	Description	Value
0x01	SCCP Flow Control=Indicates if SCCP should perform remote SCCP Status Check when routing on GT.	0x00- No (ANSI default) 0x01=Yes (ITU default)
0x02	Point Code Congestion Status=Indicates if SCCP should check the APC Congestion Status and not send a message to a congested point code.	0x00- No (ITU default) 0x01=Yes (ANSI default)
0x03	Supports the GR-246 congestion handling option. When enabled, the CSP routes the message based on the point code congestion level during GTT.	0x00 = Disabled (default) 0x01 = Enabled

SCCP SUSI (0x0067)

Purpose This section includes PPL Config Bytes and all PPL events for the PPL Component SCCP SUSI.

SCCP SUSI Configuration Bytes

Byte	Description	Value
0x01	PPL Event Indication format	0x00=Use RAW CGPA and CDPA format in PPL Event Indication. (default) 0x01=Use CGPA and CDPA Elements format in PPL Event Indication
0x03	Maximum length of data to the host in PPL Event Indications	0xF0 (default)
0x12	ICB format for indications	0x00 - Data ICB (default) 0x01 - Extended ICB format
0x14-0x17	Used to store the maximum allowable to the host for ICB length in the SCCP PPL Event Indication.	0xF0 (default)

SCCP SUSI PPL Event Indications and Requests (ITU/ANSI)

All the SCCP N-primitives are PPL events to/from PPL component SCCP SUSI.

Event ID	Event
0x01	N-UNIT-DATA
0x02	N-NOTICE (Indication only)
0x03	N-STATE
0x04	N-PC-STATE (Indication only)
0x0A	Maximum size of data to host is exceeded. The following message is truncated.

SCCP SSTC (0x006D)

Purpose This section includes Timer information for the PPL Component SCCP SSTC.

SCCP SSTC PPL Timers

Timer ID	Timer	Value
0x01	T.stat.info Timer of Periodic Sending SST Message	3000

TCAP TUSI (0x0070)

Purpose This section includes PPL Config Bytes and PPL Event Indications and Requests supported by the PPL Component TCAP TUSI.

TCAP TUSI Configuration Bytes

Byte	Description	Value
0x01	PPL Event Indication format	0x00=Use RAW CGPA and CDPA format in PPL Event Indication. (default) 0x01=Use CGPA and CDPA Elements format in PPL Event Indication
0x02	User Abort Reason to use when a TCAP Transaction Abort is initiated (due to an Matrix Controller rest or switchover)	0x00 (default)
0x03	Action taken by TCAP when a subsystem is out-of-service	0x00=Do not initiate TCAP restart when subsystem out-of-service 0x01=Initiate TCAP restart when subsystem out-of-service (default)
0x04-0x05	Maximum length of the TC-primitive ICBs in PPL Event Indications	0xFF (default)
0x06	Action for dialog when timeout occurs	0x00 Do not abort the dialog 0x01 Abort the dialog
0x07	Abort reason when dialog timeout occurs	User defined
0x08	Action to be taken when an SS7 card switchover occurs	0x00 TCAP restart is not initiated (Keep all TCAP dialogs) 0x01 Initiate TCAP restart after SS7 switchover to reset TCAP dialogs
0x09	Unused	
0x0A	Mandatory parameter validation	0x00 - Disable 0x01 - Enable (Default)

Byte	Description	Value
0x0B	Configures whether the Dialogue Terminated Parameter TLV will be included in the last TCAP indication of a dialog.	0x00 - Exclude TLV (Default) 0x01 - Include TLV
0x0C-0x11	Reserved (Do Not Use)	
0x12	ICB format for indications	0x00 - Data ICB (default) 0x01 - Extended ICB format
0x13-0xC8	Reserved (Do Not Use)	

TCAP TUSI PPL Event Indications and Requests (ITU)

All the ANSI and ITU TCAP TC-primitives are PPL events to/from the PPL component TCAP TUSI.

Event ID	Event
0x01	TC-BEGIN
0x02	TC-CONTINUE
0x03	TC-END
0x04	TC-UNI
0x05	TC-U-ABORT
0x06	TC-P-ABORT (Indication only)
0x0C	TC-RESULT-L
0x0D	TC-RESULT-NL
0x0E	TC-U-ERROR
0x0F	TC-L-CANCEL (Indication only)
0x12	TC-U-REJECT
0x13	TC-L-REJECT (Indication only)
0x14	TC-R-REJECT (Indication only)
0x15	TC-INVOKE
0x16	TC-U-CANCEL (Request only)
0x17	TC-NOTICE (Indication only)

Event ID	Event
0x18	BAD REJECT RECEIVED (Indication only)
0x1E	TCAP-RESTART
0x1F	TCAP-MESSAGE NOT COMPLETE (Indication only)
0x20	TCAP-DIALOG TIMEOUT (Indication only)
0x21	TCAP INITIATED ABORT
0x31	TC-UNI-PRIMITIVE-SET
0x32	TC-BEGIN-PRIMITIVE-SET
0x33	TC-CONTINUE-PRIMITIVE-SET
0x34	TC-END-PRIMITIVE-SET

TCAP TUSI PPL Event Indications and Requests (ANSI)

Event ID	Event
0x04	TC-UNI
0x05	TC-U-ABORT
0x06	TC-P-ABORT (Indication only)
0x07	TC-QUERY-WITH-PERMISSION
0x08	TC-QUERY-WITHOUT-PERMISSION
0x09	TC-CONVERSATION-WITH-PERMISSION
0x0A	TC-CONVERSATION-WITHOUT-PERMISSION
0x0B	TC-RESPONSE
0x0C	TC-RESULT-L
0x0D	TC-RESULT-NL
0x0E	TC-U-ERROR
0x10	TC-INVOKE-L
0x11	TC-INVOKE-NL

Event ID	Event
0x12	TC-U-REJECT
0x13	TC-L-REJECT (Indication only)
0x1E	TCAP-RESTART
0x1F	TCAP MESSAGE NOT COMPLETE (Indication only)
0x18	BAD REJECT RECEIVED FROM NETWORK
0x20	TCAP DIALOG TIME OUT (Indication only)
0x21	TCAP INITIATED ABORT (Indication only)
0x31	TC-UNI-PRIMITIVE-SET
0x32	TC-BEGIN-PRIMITIVE-SET
0x35	TC-QUERY-WP-PRIMITIVE-SET
0x36	TC-QUERY-WITHOUT PERMISSION PRIMITIVE SET
0x37	TC-CONVERSATION WITH PERMISSION PRIMITIVE SET
0x38	TC-CONVERSATION WITHOUT PERMISSION PRIMITIVE SET
0x39	TC-RESPONSE PRIMITIVE SET

TCAP CCO (0x0071)

Purpose This section includes PPL Config Bytes for the PPL Component TCAP CCO.

TCAP CCO Configuration Bytes

Byte	Description	Value
0x01-0x02	Maximum Components Length can be included in one TCAP message.	0xF0 (default)

TCAP ISM (0x0072)

Purpose This section includes PPL Config Bytes and Timer information for the PPL Component TCAP ISM.

TCAP ISM Configuration Bytes

Byte	Description	Value
0x01	Send L_CANCEL Indication upon Class 4 Operation Expiration	0x00=Disable 0x01=Enable (default)

TCAP ISM PPL Timers

Timer ID	Timer	Default Value (10 ms)
0x01	Invocation Timer	Value provided by TC_User in the TC_Invoke Primitive
0x02	Wait For Reject Timer	500

TCAP TCO (0x0073)

Purpose

This section includes PPL Config Bytes for the PPL Component TCAP TCO.

TCAP TCO Configuration Bytes

The responding Transaction ID base cannot be configured if there are active TCAP transactions in progress. If the default base (0x 01 00 00 00) must be changed, it must be done in the initialization state.

Byte	Description	Value
0x01 - 0x04	Local Transaction ID Base for Network-initiated Transactions	The default base Transaction ID is 0x01 00 00 00. To avoid confusion, it is recommended that you use a different range of Dialog IDs for TC_USER-initiated dialogs.

TCAP TSM (0x0074)

Purpose This section includes PPL Config Bytes for the PPL Component TCAP TSM.

TCAP TSM Configuration Byte

Byte	Description	Value
0x05	Determines whether to use the CGPA and CDPA in the first responding TC_END request as the message returning address. This configuration byte applies to the case when the host sends the first response message (TC_END req) responding to the incoming TC_BEGIN.	0x00 - The addresses in the incoming BEGIN is used as the returning address. The CGPA and CDPA in the TC_END req is not used. 0x01 - The CGPA and CDPA in the TC_END req (if included) are used as the returning address for the outgoing END message.

TCAP DHA (0x0075)

Purpose This section includes a new Timer for the PPL Component TCAP DHA (Dialog Handling). The TCAP dialog timeout timer is measured as "how long the dialog is idle." In other words, how long there are no messages exchanged for a dialog.

TCAP DHA PPL Timers

Timer ID	Timer	Default Value
0x01	TCAP Dialog timeout threshold	64800 seconds

8 ISDN

Purpose This chapter describes the Dialogic implementation and features of Integrated Services Digital Network (ISDN), Primary Rate Interface (PRI). References to ISDN functionality in this chapter involve both the ISDN PRI card and the ISDN Series 3 card.

Introduction to ISDN

Overview

The Dialogic Integrated Services Digital Network (ISDN) product provides multiple communications tasking (voice, data, compressed),128 Kbps high-speed and high-bandwidth service. ISDN, non-compressed (512 KBps compressed) also provides on-demand service, connects up to eight devices simultaneously, and can be call-managed. The basic configuration is 23+D (Primary Rate Interface) and 30B+D outside of North America.

Description

The ISDN card, in conjunction with the T1 and E1 network interface cards, interfaces to various equipment types supporting the ISDN PRI protocol. These include tandem (Class 4) switches, end office (Class 5) switches, PBXs, and proprietary implementations, not only in North America but throughout the world.

Implementation

The Dialogic ISDN PRI implementation is based on ITU-T Q.921 and Q.931 specifications. Each implemented variant references the appropriate interface document supplied by the equipment manufacturer. The interface document is usually a variant of the ITU-T recommendations.

Since each manufacturer supports different services, the host manages the information to send and receive in specific messages. This lets you access many features and services at the host layer and carry them out using PPL-controlled call control logic.

Each ISDN card supports 32 D channels. Each D channel provides the High Data Link Control (HDLC) communications over T1 or E1 on one timeslot on the span. Each D channel can control up to 19 other spans in addition to the span on which it is located.

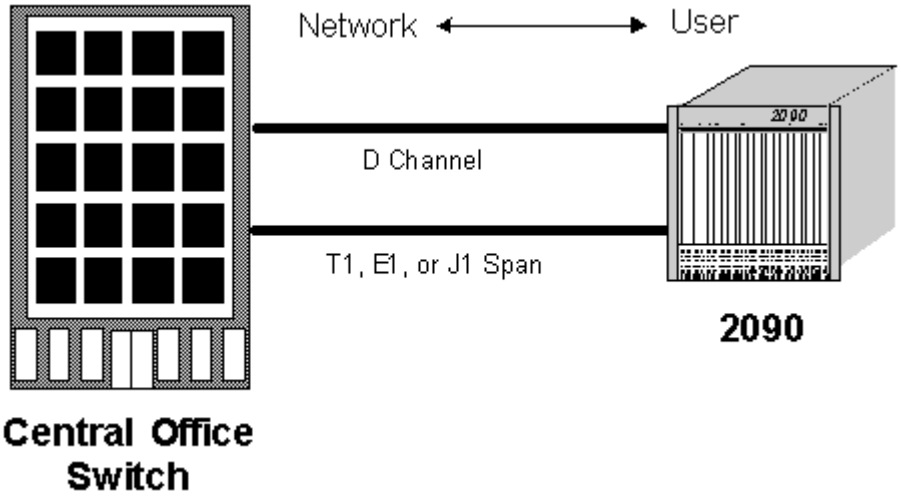
National ISDN PRI NI 2

The CSP system software ISDN stack supports National ISDN PRI NI 2. National ISDN PRI supports Non-Facility Associated Signaling (NFAS), allowing up to 30 DS1 interfaces and the D channel backup procedure, as well as B channel availability and provisioning. Connection endpoint variants are defined for National ISDN User side and Network side.

Diagram

The figure below shows an ISDN Point-to-Point Connection with an CSP connected to a central office switch.

Figure 8-1 ISDN Point-to-Point Connection



Customization

Depending upon the provisioning requirements, you can have one D channel per span [Facility Associated Signaling (FAS)], or one D channel managing up to 20 spans [Non-Facility Associated Signaling (NFAS)]. You can control a total of 32 spans using all FAS D channels, or up to 20 spans per NFAS D channel (to a maximum of 64 spans per system). You can intermix both FAS and NFAS D channels on a card with each supporting a different variant. NFAS spans can be configured dynamically during call processing.

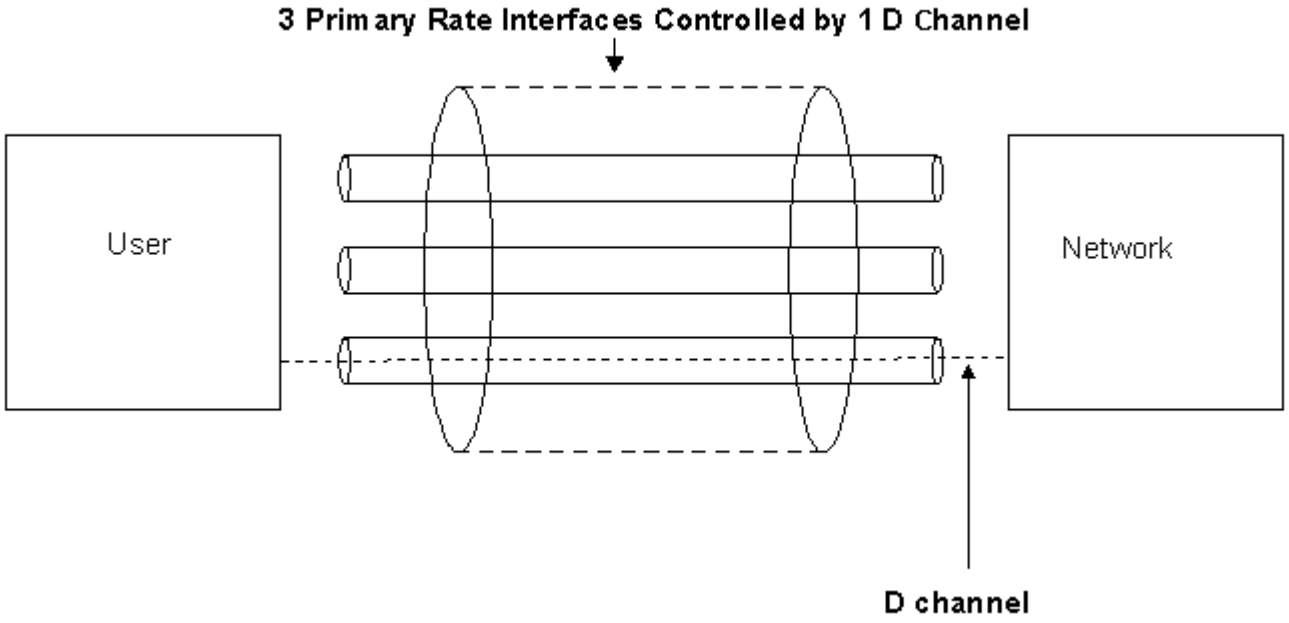
You can insert multiple ISDN cards in any line card slot in the CSP. Each has unrestricted access to all timeslots, so you can configure any timeslot in the system as a D channel.

NFAS and FAS Examples

When using NFAS, the D channel on one span controls the B channels on other spans. However, with NFAS a backup D channel is assigned for redundancy on a separate span.

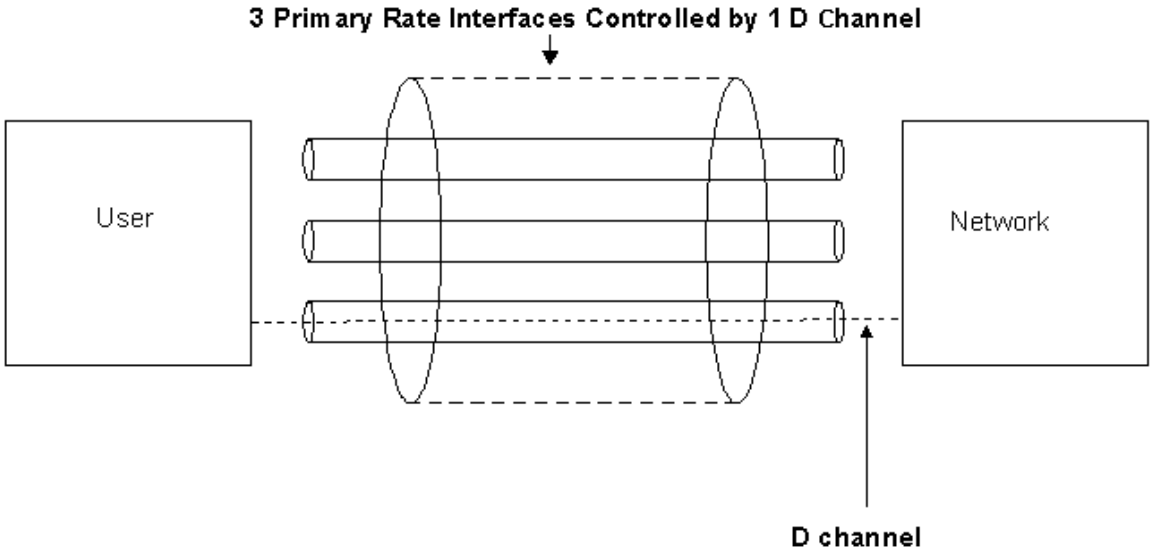
The figure below shows an example of NFAS Controlling three Primary Rate Interfaces (PRI).

Figure 8-2 Three Primary Rate Interfaces Controlled by 1 D channel



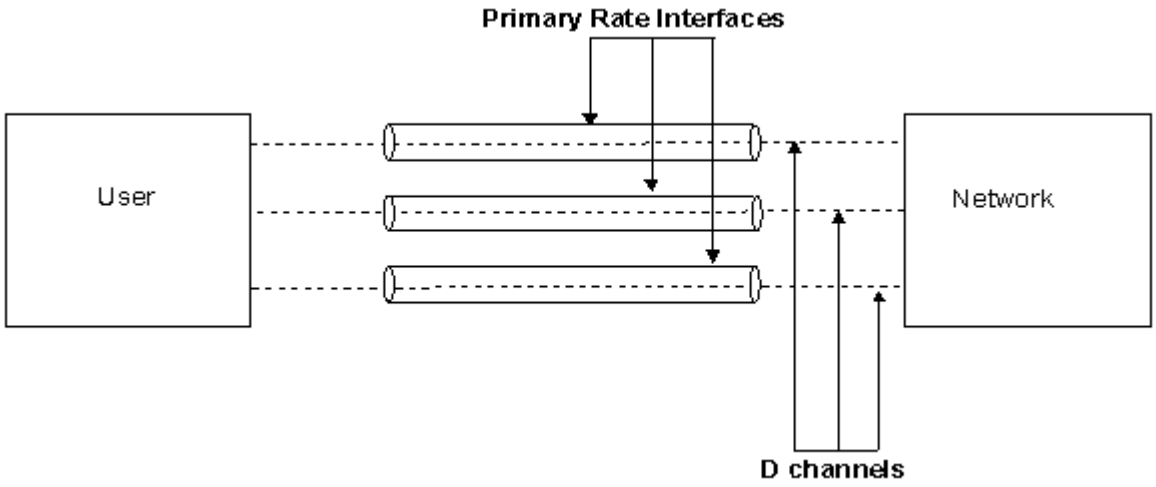
The figure below shows an example of NFAS with Backup D Channel Controlling three PRIs.

Figure 8-3 Three Primary Rate Interfaces Controlled by 1 backup D channel



When using FAS there is a D channel on the same span as a B channel that is controlled by the D channel. The figure below shows an example of FAS on each of three PRIs:

Figure 8-4 One D channel controls one B channel



Supported Features

ISDN cards supports the following:

- 32 D channels per-card over T1 or E1
- ITU-T Q.921, Q.931-based
- Euro-ISDN (Includes French and German Delta), JATE (INS 1500), Lucent 4ESS Q.931 PRI, Lucent 5ESS Q.931 PRI (Custom), Northern Telecom DMS-100 Q.931 PRI (Custom), Northern Telecom DMS-250 Q.931 PRI (Custom), and AUSTEL.
- Network Side Euro-ISDN, Network Side ISDN BRI - Lucent 4ESS
- D channel backup
- NFAS (Not supported over E1)
- Incoming B channel Negotiation
- Flexible Information Element (IE) reporting and parsing (with PPL messages)

- Card Level Redundancy

Redundancy

After Installing two ISDN cards with their appropriate I/O cards redundancy is provided for all 32 D channels. If the active card fails, or if you remove the card, the standby card takes over the call processing operations. All calls in a connected state are retained while others are purged.

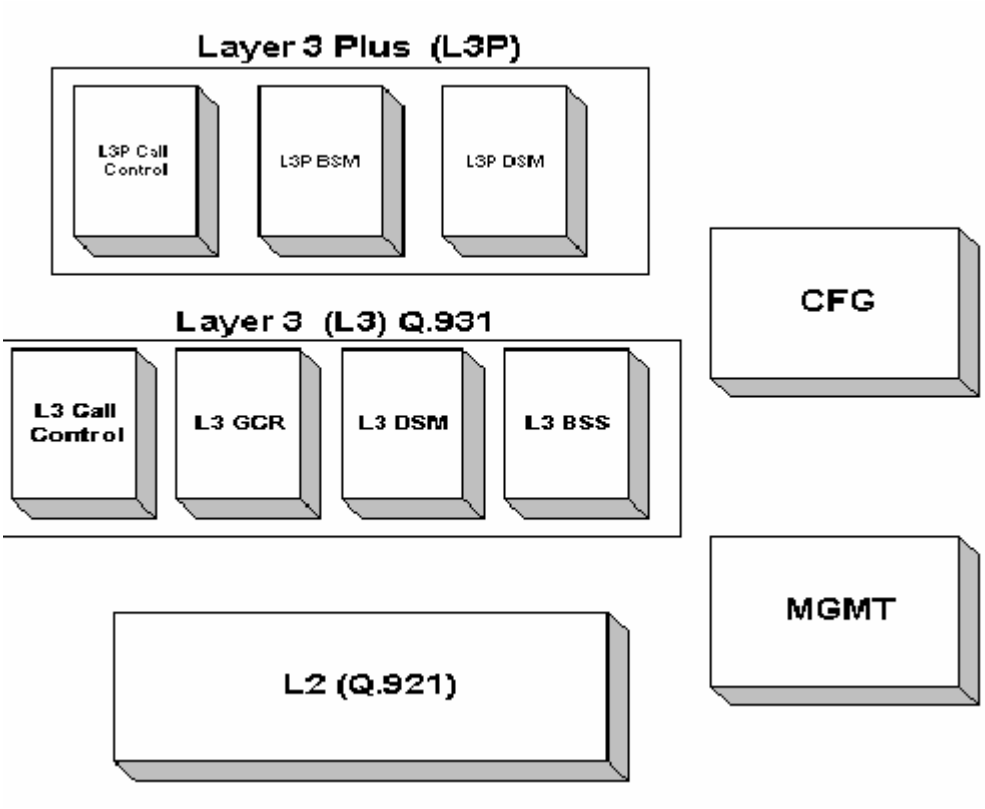
ISDN Architecture

Overview This section describes the modules that form the ISDN PRI software architecture. These modules are:

- Layer 3 Plus
- Layer 3
- Layer 2
- Configuration
- Management

Diagram Figure 8-5 illustrates the functional modules involved in the PRI implementation on an ISDN card.

Figure 8-5 ISDN PRI Software Architecture



Layer 3 Plus

The Layer 3 Plus (L3P) module is an interface between Layer 3 and Layer 4 Central Call Processing (CCP) on the Matrix Controller, and Layer 5 (host). L3P formats information from Layer 3 into the programmed format for the host. It also manages application specific variants for call control. It is made up of the following PPL components:

L3P Call Control [Component (0x05), per D channel]

- This component is the interface between Layer 3 and Layer 4 central call processing, which resides on the Matrix Controller. This component manages which messages to send internally for call processing as well as notifying the host of various events related to the protocol.

L3P D Channel Control [Component (0x06), per D channel]

- This component manages the enabling/disabling of the D channel per terminal (for PRI ISDN there is only one terminal). It manages the sending of the Alarm message when the D channel is establishing a connection, and informs the L3P B channel control component of the availability of D channel.

L3P B Channel Control [Component (0x07), per B channel)

- This component manages the service state of the channel based upon the following:
  - Host service states
  - Layer 1 states (span framed or not)
  - D channel availability
  - B channel availability from the network

Layer 3

The Layer 3 (L3) module provides ITU-T Q.931 implementation. The L3 state diagrams mirror the SDL diagrams of the interface specifications. It consists of the following PPL components:

L3 Call Control [Component (0x08) per D channel]

- This component implements Q.931 variants, which has state tables that are SDL equivalents with some validation logic embedded. This component is managed per call reference, and manages B channel bearer capability as well as B channel allocations.

L3 Global Call Control [Component (0x09) per D channel]

- This component manages the protocol for the restarting logic. It implicates active calls per a channel RESTART message, ACKs them, and then arbitrates sequencing with the B channel control components.

L3 D Channel Control [Component (0x0A) per D channel]

- This component manges the D channel availability as defined by the Data Link Connection state diagram in ITU-T Q.931. It communicates with the L3P DSM upon D channel failure.

L3 B Channel Control [Component (0x0B) per B channel]

- There is one state machine per B channel to manage the SERVICE messages for the protocol using the protocol discriminator (0x03). This state machine manages availability for the B channels as well as in-use status.

Layer 2 The Layer 2 (L2) module is the ITU-T Q.921 (LAPD) implementation. It performs flow control, sequence numbering (module 128), retry logic, and it also guarantees message delivery.

Configuration The Configuration (CFG) module manages all configuration of parameters and options for the ISDN protocol. This task communicates with the Matrix Controller during configuration. It also verifies battery backed data and card level configuration.

Management The Management (MGMT) module performs terminal endpoint identifier (TEI) management logic for Basic Rate Interface (BRI) access and it synchronizes Layers 2 and 3 correctly. It also acts as an alarm manager for the L2 protocol.

ISDN Software Features

Overview

ISDN cards support the following features:

- D Channel Backup
- Information Element Library
- B Channel Negotiation
- ANI on Demand
- Vari-A-Bill Billing

D Channel Backup

ISDN cards support the D channel backup procedure as defined by Lucent TR 41459. This procedure provides a standby D channel when using NFAS. When an active D channel fails, or if there is a span alarm, the CSP converts to the standby D channel and maintains active calls. This is a provisioned option and is applicable only when using NFAS. D channel backup is supported on the T1 cards only.

ISDN Information Elements

Dialogic provides a format to send and receive Information Elements (IEs) per D channel. The data is represented using the Information Control Block (ICB) format with the ICB subtype of “Formatted IEs” (see *Information Control Blocks* chapter in the *API Reference*). This allows greater flexibility when sending a message that has different types of data to transport. Multiple Formatted IEs can be loaded in the data section of the ICB.

B Channel Negotiation

ISDN cards support B channel negotiation on incoming calls as defined in Lucent TR41459 for both E1 and T1 cards. This feature allows the user receiving the incoming call (SETUP) to negotiate for the selection of the B channel. Only B channels controlled by the same D channel are considered for the selection procedure.

ANI On Demand

ISDN cards support the Automatic Number Identification (ANI) on Demand feature as specified in Lucent TR41459. This feature allows the host to request the call originating number (calling party number) from networks that support ANI such as AT&T MultiQuest. A *PPL*

Event Request of ANI on Demand (0x21) must be sent from the host immediately after receiving the *Request For Service* message from the CSP.

Vari-A-Bill Billing

ISDN cards support the Vari-A-Bill feature as specified in Lucent TR41459. This feature allows the host to request a change in billing for an incoming call. You can use this feature after a call has been answered and only if the SETUP message indicates that the network supports the flexible billing service.

Configuring ISDN PRI

Purpose

This section describes the procedures for configuring primary rate ISDN and National ISDN PRI NI 2. The first four steps of the basic ISDN configuration are described in more detail in other parts of this chapter:

- Span Configuration
- D Channel Configuration
- B Channel Configuration
- Optional Configuration

More details about bringing spans and channels in service, reconfiguration, and querying information are provided following the configuration sequences in this section.

Before you begin

The basic ISDN configuration assumes a 23+D (Primary Rate Interface) in North America and 30B+D outside of North America.

ISDN PRI Configuration Sequence

The table below shows the basic ISDN configuration sequence.

Step	Action	Description	API Message
1	Configure Spans	- Assign Logical Span IDs - Configure span formats	Assign Logical Span ID T1 Span Configure E1 Span Configure
2	Configure the D channels	- Assign D channels - Configure D channels - Add NFAS Facilities	D Channel Assign ISDN Interface Configure D Channel Facility List Configure
3	Configure the B channel	Configure B channel options	B Channel Configure
4	Optional Configuration	- PPL Customization - Features	
5	Bring spans and channels in service	Bring spans, D channels, and B Channels in service	Service State Configure

National ISDN User Side Configuration

The following table provides a sample procedure for configuring the User Side endpoint variant of National ISDN PRI NI 2:

Step	Action
1	De-assign the Logical Span ID of the ISDN card by setting the slot and spans (logical and physical) to 0xFF. Use the Assign Logical Span ID (0xA8) message.
2	Assign logical span IDs to spans and channels. Use the Assign Logical Span ID (0xA8) message.
3	Configure the T1 Span. Use the T1 Span Configure (0xA9) message.
4	Assign a channel as an ISDN PRI D channel. Use the D Channel Assign (0xC4) message.
5	Define the connection type for National ISDN PRI NI 2 user side as 0x09. Use the ISDN Interface Configure (0x60) message.
6	Define the spans that are controlled by a D channel (including any spans in NFAS mode). Use the D Channel Facility List Configure (0xC6) message. For the Action field, select 0x01 (Add Facility) For Facility Number, select 0x01-0x1E, depending on the number of spans being added.
7	Define the network type for the B channels. Use the B Channel Configure (0xC8) message. For the Network Type (0x01) field, select 0x01 (Do Not Include Network-Specific Facilities IE)
8	Bring spans, B channels and D channels in-service. Use the Service State Configure (0x0A) message with appropriate AIBs. Send the message for each configuration.

Bringing Spans and Channels In Service

When all of the configuration is complete, bring up the D channel to establish a connection with the network and then begin call processing.

After establishing a connection to the network, the CSP sends the *DS0 Status Change* message to the host with the status of in-service. The *DS0 Status Change* messages follow for the B channels (voice/data channels).

If there is a link failure, the CSP sends the *DS0 Status Change* message to the host for the D channel, as well as all of the associated B channels. All channels have a status of out-of-service.

Reconfiguration

Before performing reconfiguration, the CSP must be in a known state regardless of system events such as the removal or reset of cards or a host restart. Depending upon the event, the CSP determines which steps to take to get the interface operational.

The simplest approach is to send a *Reset Configuration* message indicating 0xFF for the Matrix Controller slot number. The configuration of all cards resets. The application must wait for the *Card*

Status Report messages to be sent to the host before configuration begins. Sending this message clears all host-configured options, including downloaded PPL tables.

Another approach is to use *Reset Configuration* for the ISDN slot number, which defaults configuration for all D channels assigned to the card. Use this approach for an application that reconfigures certain features in the CSP during a live installation.

To remove a single D channel's assignment and configuration, send the *Assign Logical Span ID* message (indicating de-assignment) for the D channel span.

Querying Information

Use the *ISDN Query* message to get parameters configurable with the *ISDN Terminal Configure* and *ISDN Interface Configure* messages, as well as assigned protocols.

Example

The following is an example of the *ISDN Query* message to query the General Interface Options and the response from the CSP.

API Message

```
Trace H->X FE 00 0E 00 63 00 00 01 00 01 0D 03 00 01 17 01
   00
```

Byte	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x000E
3, 4	Message Type	0x0063
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Data Length	0x03
12, 13	Data[0,1] Logical Span ID	0x0001
14	Data[2] Channel	0x17 (Channel 23)
15	Query Type	0x01 (General Interface Options)
16	Query Subtype	0x00 (None)
170	Checksum	0xCS (not shown in trace)

API Response

```
Trace X->H FE 00 1B 00 63 00 00 01 00 10 00 01 00 00 00 01
   00 00 00 01 00 01 00 01 00 01 00 03 00 00
```

Byte	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x001B

Byte	Field Description	Value and Indication
3, 4	Message Type	0x0063
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
10, 11	Data[0,1] Connection Type	0x0001 (Lucent 4ESS)
12, 13	Data[2,3] Options	0x0000 (No options)
14, 15	Data[4,5] D Channel Physical Medium	0x0001 (64 Kbps)
16, 17	Data[6,7] HDLC Bit Polarity	0x0000 (Normal)
18, 19	Data[8,9] Network Side Layer 2	0x0001 (Network Side (C/R Bit Inverted))
20, 21	Data[10,11] B Channel Selection Mode	0x0001 (Linear Clockwise)
22, 23	Data[12,13] Location	0x0001 (Private Network/Local User)
24, 25	Data[14,15] Channel Release Request on ISDN Disconnect	0x0001 (Send Host Channel Release Request on ISDN Disconnect)
26	Data[16,17] Protocol Discriminator Value for Maintenance Messages	0x0003 (Default)
28	Data[18,19] B Channel Encoding for Transmission	0x0000 (Channel Number)
30	Checksum	0xCS (not shown in trace)

Network Side Euro-ISDN

Overview

The ISDN software module supports network side Euro-ISDN PRI variant. This allows CSPs to handle basic call control, interfacing with a user-side.

ETSI specification support

The following is supported as defined in the applicable ETSI specification:

- All procedures, messages, and message content defined as mandatory for network side functionality.
- All call states supported in Dialogic's ISDN PRI user side implementation.
- All network side call setup and tear-down procedures.
- All mandatory, bidirectional (network-to-user and user-to-network), circuit mode connection control messages.
- All mandatory network-to-user messages.
- All information elements that are designated as mandatory for network-to-user side messages.
- All timer values.
- All network side error handling procedures.

Supported Q.931 Messages

The following Q.931 messages are supported:

Call establishment

- ALERTING
- CALL PROCEEDING
- CONNECT
- CONNECT ACKNOWLEDGE
- PROGRESS
- SETUP
- SETUP ACKNOWLEDGE

Call information phase messages

- USER INFORMATION

Call clearing messages

- DISCONNECT

- RELEASE
- RELEASE COMPLETE

Miscellaneous

- FACILITY
- INFORMATION
- NOTIFY
- STATUS
- STATUS ENQUIRY

Other

- REGISTER

Specification Compliance

The ISDN PRI network side implementation complies to the signals and signaling procedures described by the ETSI 300 102-1 and 300 102-2 specifications specific to network side, with the following exceptions:

Dialogic's network side Euro-ISDN PRI implementation does not support the following:

- Call re-arrangement procedures (SUSPEND and RESUME) (ETSI 300 102-1 section 5.6)
- Message segmentation (ETSI 300 102-1 section 5)
- Sending the STATUS ENQUIRY message (ETSI 300 102-1 section 5.8.10)
- Sending the CONGESTION CONTROL message (ETSI 300 102-1 section 3.1.3)
- Sending the Keypad Facility Information Element

Configuration

The network side ISDN PRI configuration procedure is the same as for the user side except that you must configure the D Channel for Network Side using the ISDN Interface Configure 0x0060 message (see the API message formats in the *API Reference*).

Example Message

The following is an example of the *ISDN Interface Configure* message, which changes the Connection Type to network side Euro-ISDN.

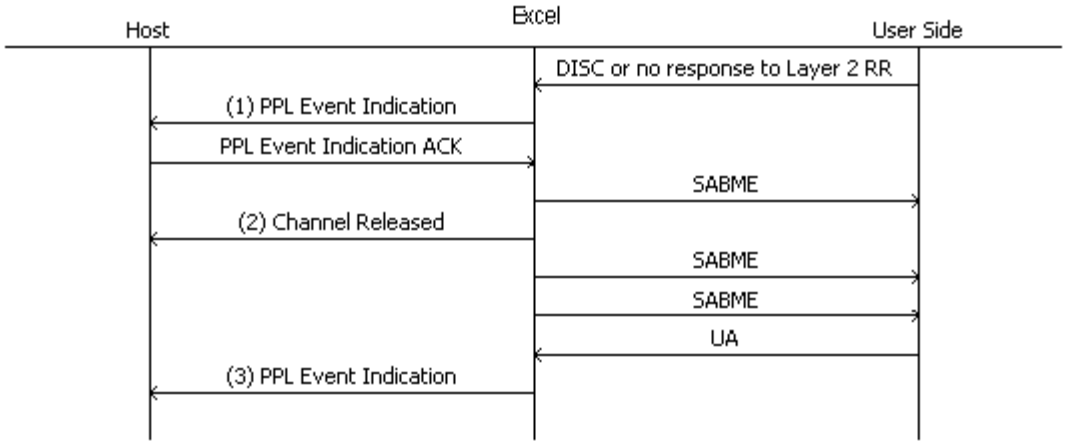
```
Trace H->X FE 00 0F 00 60 00 00 01 00 01 0D 03 00 00 17 01
   00 17
```

Byte	Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x000F
3, 4	Message Type	0x0060
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Data Length	0x03
12, 13	Data[0,1] Logical Span ID	0x0000 (Span 0)
14	Data[2] Channel	0x17 (Channel 23)
15	Entity	0x01 (Connection Type)
16, 17	Data[0,1] Value	0x0017 (Network Side Euro-ISDN)
18	Checksum	0xCS (not shown in trace)

Call Flows

D Channel Failure with Re-establishment of D Channel

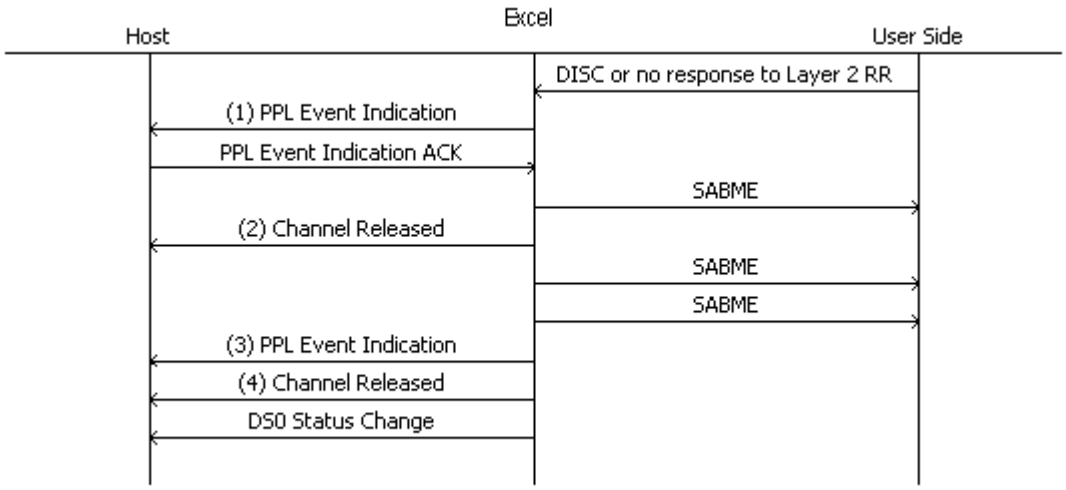
The following figure shows a call flow for a D channel failure followed by the re-establishment of the D channel.



1. Upon receiving a Layer 2 DISCONNECT, or no response to a Layer 2 RR (T203 expiry), from the User Side indicating a D channel failure, the CSP sends a *PPL Event Indication* message of Network Side D Channel Temporary Malfunction (0x04).
2. All non-active calls are released and the host is sent *Channel Released* messages for each channel.
3. If the D channel is re-established within the period of the T309 timer (90 s), all active calls are maintained and the host is sent a *PPL Event Indication* message of Network Side D Channel Re-established (0x06.)

D Channel Failure with no D channel Re-establishment

The following figure shows a call flow for a D channel failure with no re-establishment of the D channel.



1. On receiving a Layer 2 DISCONNECT, or no response to a Layer 2 RR (T203 expiry), from the user side indicating a D channel failure, the CSP sends a PPL event of Network Side D Channel Temporary Malfunction (0x04).
2. All non-active calls are released and the host is sent *Channel Released* messages for each channel.
3. If the D channel is not re-established within the T309 period (90 s), the host is sent a *PPL Event Indication* message of Network Side D Channel Down (0x05).
4. All active calls are released and a *Channel Released* message is sent to the host for each channel, followed by:
   - A *DS0 Status Change* message of Out of Service for all channels.

Span Configuration

Overview

This section describes the first step in the basic ISDN configuration sequence, span configuration.

Span configuration includes:

- Assigning Logical Span IDs
- Configuring the Span Format and Line Length

Assigning Logical Span IDs

First, de-assign spans you want to use for ISDN with the *Assign Logical Span ID* message to clear all D channels and facility assignments. De-assigning a logical span ID results in taking the span and all associated channels and facilities that are assigned out-of-service. Then, assign Logical Span IDs to spans to the spans with the *Assign Logical Span ID* message.

Configuring Span Formats

Set the format of the span on which the ISDN D channel resides using the *T1 Span Configure* or *E1 Span Configure* messages

Configure the framing and line coding for all ISDN PRI spans as follows:

T1

- Framing - ESF
- Line Coding - B8ZS
- Signaling - Clear Channel

E1

- Coding - HDB3
- Signaling - Clear Channel

Important! If an E1 span is not configured for Euro-ISDN, the encoding format for all ISDN E1 channels defaults to u-law. You must change the encoding format with the *PCM Encoding Format Configure* message prior to bringing channels into service. If the format is not changed, DSP resource requests will result in a NACK of *DSP Not Configured For Requested Function* (0x27).

Euro-ISDN E1

- Coding - HDB3
- Error Checking - Enable CRC (bit 1) and FEBE (bit 2)

- Signaling - Clear Channel (bit 3)/E1 Layer 1 Management (bit 5)
- Line Length - G.703.

Austel-ISDN

- Coding - HDB3
- Error Checking - Enable CRC (bit 1) and FEBE (bit 2)
- Signaling - Clear Channel (bit 3)/E1 Layer 1 Management (bit 6)
- Line Length - G.703.

ISDN D Channel Configuration

Overview

This section describes the second step in the basic ISDN configuration sequence.

- Assigning D channels
- Configuring D Channel options
- Adding facilities (for NFAS)

Assigning D Channels

The *D Channel Assign* message allocates a D channel at the timeslot you specify. In order for this assignment to succeed, the network interface card on which the D channel is to reside must be in service. The physical span MUST be assigned. You must specify the slot number of the ISDN card, as well as the ISDN type (primary and secondary).

> **Important!** The *D Channel Assign* message sets all B channel configuration parameters back to their defaults.

The CSP software selects the actual D channel resource. If the host attempts to assign a D channel to a card that already has 32 D channels assigned, the host receives a Response Status of “ISDN D Channel Exceeds Max” (0xA4). You must indicate another card for D channel processing.

All subsequent references to the D channel timeslot use the Logical Span ID and channel specified in the message. The facility number of 0 (zero) is defaulted to the span with the D channel assignment. When assigning the secondary type, the host selects the facility number.

Requirements for Assigning a D Channel

The following are requirements for assigning D channels:

- Span must be assigned.
- Network interface card(s) must be present and operational.
- ISDN card must be operational and have available D channel(s).
- Span and channel must not be assigned to an ISDN D or B channel location. For most applications, the D channel resides on the 24th channel of the T1 span, and timeslot 16 over E1.

Example

The following is an example of the *D Channel Assign* message. The example assumes that the Logical Span ID 0 is assigned and that the ISDN card is in slot 4.

```
Trace H->X FE 00 0E 00 C4 00 00 01 00 01 15 04 04 00 00 17
   00
```

Byte	Field Description	Value and indication
0	Frame	0xFE
1, 2	Length	0x000E
3, 4	Message Type	0x00C4
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Type	0x15 (Primary D Channel)
11	Data Length	0x04
12	Data[0] Slot #	0x04
13, 14	Data[1] Logical Span ID	0x0000
15	Data[3] Channel	0x17 (Channel 23)
16	Secondary D Channel Facility Number	0x00
17	Checksum	0xCS (not shown in trace)

Configuring D channels

The *ISDN Interface Configure* message configures attributes required for ISDN connections to the public network. To configure the connection endpoint variant, determine the type of switch from which the D channel will originate. This requires consultation with the provider. The default switch type is the Lucent 4ESS.

Normally, you do not need to configure Layer 2 and Layer 3 timers and counters. However, if it is necessary to change these, refer to ITU-T Q.921 and Q.931 for details on what they do and represent.

The table below lists the ISDN interface entities and their defaults. Use the *ISDN Interface Configure* message to change the default configuration.

Option	Default
Connection Type	Lucent 4ESS
D Channel Physical Medium	64 kbps
HDLC Bit Polarity	Normal
Network Side Layer 2	User Side
B Channel Selection Mode	Linear Clockwise
Location	User
Layer 4 Channel Release Request on ISDN DISCONNECT	Disabled
Protocol Discriminator Value for Maintenance Messages	3
B Channel Encoding for Transmission	Channel Number

Requirements for Configuring D Channel Options

The following are requirements for configuring D channel options:

- The D channel must be assigned.
- The D channel parameters will only be initiated when the transition from out-of-service to in-service occurs.

Reconfiguring D Channels

Follow the steps below to reconfigure a D channel:

1. Take the B Channels out of service with the *Service State Configure* (0x000A) message.
2. Take the D Channels out of service with the *Service State Configure* (0x000A) message.
3. De-assign the D Channel with the *D Channel De-assign* (0x00C5) message.
4. Reconfigure the D channel with the *D Channel Assign* (0x00C4) message.

5. Reconfigure general options to an ISDN interface with the *ISDN Interface Configuration* (0x0060) message.
6. Reconfigure B channel information with the *B Channel Configure* (0x00C8) message
7. Bring span(s), B Channels and D channel(s) back into service with the *Service State Configure* (0x000A) message.

L3 Default Settings per Connection Type

The table below lists the default L3 configuration values for all connection types currently supported.

Connection Type			Default Settings	
Lucent 4ESS (0x01) Lucent 5ESS (0x02)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 15 s (D Backup) or 30 s N200 = 3	N201 = 260 ACK Pending timer = 0.5 s k = 7 L2 Options = 0
Lucent 4ESS (0x01) Lucent 5ESS (0x02)	PPL Timers	L3 Call Control Component (0x08)	T303 = 4 s T305 = 4 s T308 = 4 s T310 = 25 s T313 = 4 s	
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03	#5 = False #6 = False #7 = 1 (Number of setup resends) #10 = False
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	Other		Restart Procedure initiated	
DMS100/250 (0x03/0x04)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 30 s or 15 s (D Backup) N200 = 3 s	N201 = 260 ACK Pending timer = 0.5 s k = 7 L2 Options = 0

Connection Type			Default Settings	
DMS100/250 (0x03/0x04)	PPL Timers	L3 Call Control Component (0x08)	T303 = 4 s T305 = 4 s T308 = 4 s T310 = 10 s	T313 = 4 s
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03	#5 = False #6 = False #7 = 1 #10 = False
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	Other		Restart initiated	
AUSTEL (0x05) AUSTEL Netside (0x15)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 10 s N200 = 3 s N201 = 260	ACK Pending timer = 0.35 s k = 7 L2 Options = 0x02 (stop sending SABME after N200)

Connection Type			Default Settings	
AUSTEL (0x05) AUSTEL Netside (0x15)	PPL Timers	L3 Call Control Component (0x08)	T304 = 15 s T305 = 30 s T308 = 4 s T313 = 4 s	
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03	#5 = False #6 = False #10 =False
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	PPL Config Bytes	L3 D Channel Control Component (0x0A)	#1 = True	
	B Channel Options		Network Specific = 0x01 (no NSF) Calling Number Type = 0x01 (unknown) Called Number Type = 0x01 (unknown)	
	PCM Encoding Format		a-law	
	Other		Restart procedure is not supported. The ISDN card automatically responds to a valid SETUP message with both a CALL PROCEEDING and an ALERTING message.	
JATE (0x06)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 30 s N200 = 3 s N201 = 260	ACK Pending timer = 0.5 s k = 7 L2 Options = 0

Connection Type			Default Settings	
JATE (0x06)	PPL Timers	L3 Call Control Component (0x08)	T303 = 4 s T305 = 4 s T308 = 4 s T310 = 30 s T313 = 4 s	
	PPL Timers	L3 Global Call Reference Component (0x09)	Timer 16: T316 = 120 s	
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03	#5 = False #6 = False #7 = 1 #10 =False
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	PPL Config Bytes	L3 D Channel Control Component (0x0A)	#1 = False	
	B Channel Options		Network Specific = 0x01 (no NSF) Calling Number Type = 0x01 (unknown) Called Number Type = 0x01 (unknown) Calling Number Plan = 0x01 (unknown) Called Number Plan = 0x01 (unknown)	
	PCM Encoding Format		a-law	
	Other		Restart procedure is supported For an outgoing call, a 55-second timer is started after the ALERTING message is received. If the timer expires before a CONNECT message is received, the call is cleared (a DISCONNECT message is sent).	
EURO (0x07)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 10 s N200 = 3 s N201 = 260	ACK Pending timer = 300 ms k = 7 L2 Options = 0

Connection Type			Default Settings	
EURO (0x07)	PPL Timers	L3 Call Control Component (0x08)	T301 = 180 s T302 = 15 s T303 = 4 s T304 = 15 s T305 = 30 s	T308 = 4 s T310 = 25 s T313 = 4 s T322 = 4 s
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03 #5 = False	#6 = False #7 = 1 #8 = False #9 = False #10 = False #11 = False #22 =3
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	PPL Config Bytes	L3 D Channel Control Component (0x0A)	#1 = False	
	B Channel Options		Network Specific = 0x01 (no NSF) Calling Number Type = 0x01 (unknown) Called Number Type = 0x01 (unknown)	
	PCM Encoding Format		a-law	
	Other		Restart procedure is not supported	
NI2 User (0x09)	See NI2 Netside (0x19)			
EURO NETSIDE (0x17)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 10 s N200 = 3 s N201 = 260 s	ACK Pending timer = 300 ms k = 7 L2 Options = 0

Connection Type			Default Settings	
EURO NETSIDE (0x17)	PPL Timers	L3 Call Control Component (0x08)	T302 = 15 s T303 = 4 s T304 = 20 s	T305 = 30 s T308 = 4 s T310 = 25 s
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03 #5 = False	#6 = False #7 = 1 #8 = False #10 = False #11 = False
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	PPL Config Bytes	L3 D Channel Control Component (0x0A)	#1 = False #2 = True Enable T309 logic	
	PPL Timers	L3 D Channel Control Component (0x0A)	T309 = 90 s	
	PCM Encoding Format		a-law	
	Other		Restart procedure is not supported.	
NI2 User (0x09) NI2 Netside (0x19)	ISDN Terminal Configure	LAPD Entities	T200 = 1 s T203 = 15 sec or 30 sec (D backup) N200 = 3 N201 = 260	ACK Pending timer = 0.5 s k = 7 L2 Options = 0x08 (Enable Frame Reject Response (FRMR) sending.)

Connection Type			Default Settings	
NI2 User (0x09) NI2 Netside (0x19)	PPL Timers	L3 Call Control Component (0x08)	T301 = 5 m T303 = 4 s T304 = 15 s T305 = 4 s T308 = 4 s	T310 = 25 s T313 = 4 s T322 = 4 s
	PPL Config Bytes	L3 Call Control Component (0x08)	#1 = True #2 = False #3 = False #4 = 0x03	#5 = False #6 = False #10 =False #7 = 0x01 #8= 0x2C #22=0x03
	PPL Timer	L3 Global Call Reference Component (0x09)	T316 = 2 m	
	PPL Config Bytes	L3 Global Call Reference Component (0x09)	#1 = False #2 = 2 #3 = False #4 = False #5 = False	#6 = False #7 = False #8 = False #9 = False #10 = False
	PPL Config Bytes	L3 D Channel Control Component (0x0A)	#1 = False #2 = True	
	PPL Timers	L3 D Channel Control Component (0x0A)	T321 = 40 s (user) T321 = 50 s (netside) T309 = 90 s	

Example Message

The following is an example of the *ISDN Interface Configure* message.

```
Trace H->X FE 00 0F 00 60 00 00 01 00 01 0D 03 00 00 17 01
   00 01
```

Byte	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x000F
3, 4	Message Type	0x0060
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Data Length	0x03
12, 13	Data[0,1] Logical Span ID	0x0000 (Span 0)
14	Data[2] Channel	0x17 (Channel 23)
15	Entity	0x01 (Connection Type)
16, 17	Data[0,1] Value	0x0001 (Lucent 4ESS)
18	Checksum	0xCS (not shown in trace)

Adding Facilities

The *D Channel Facility List Configure* message adds or deletes facilities controlled by a D channel. For FAS arrangements, facility 0 is assumed for the span with the D channel. By sending the *D Channel Facility List Configure* message, NFAS is automatically assumed.

Important! NFAS is only supported over T1, not E1.

Specify the span and channel for D and assign the facility number (1-19 for NI2 and 1-9 for other channel types) and the span ID for the facility being assigned. Each Primary D channel assumes it is located on facility 0. You can add up to 19 facilities with the *D Channel Facility List Configure* message.

The facility number is the span offset for the controlling D channel. For example, for NFAS the span on which the D channel is located is facility number 0. The next span controlled by that D channel is facility # 1, the next is facility # 2, and so on.

Requirements for Adding Facilities

The following are requirements for adding facilities to a D channel:

- D channel must be assigned (use the *D Channel Assign* message).
- Facility Number must be from 1-19 (0 is the D channel facility number).
- Facility must be assigned (use the *Assign Logical Span ID* message).
- Span and channel must not currently be assigned to an ISDN D or B channel location. For most applications, the D channel resides on the 24th channel of the T1 span.

Example

The following example shows the *D Channel Facility List Configure* message adding Logical Span ID #2, and assigning it to facility #1 for D channel 00,17.

Trace H->X FE [00 12] [00 C6] 00 04 01 00 02 0D 03 00 00 17 0C 02 00 02 01 01]

Byte	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x0012
3, 4	Message Type	0x00C6
5	Reserved	0x00
6	Sequence Number	0x04
7	Logical Node ID	0x01

Byte	Field Description	Value and Indication
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x02
10	Address Element 1 Address Type	0x0D (Channel)
11	Data Length	0x03
12, 13	Data[0,1] Logical Span ID	0x0000
14	Data[2] Channel	0x17
15	Address Element 2 Address Type	0x0C (Span)
16	Data Length	0x02
17, 18	Data[0,1] Logical Span ID	0x0002
19	Action	0x01
20	Facility Number	0x01
21	Checksum	0xCS (not shown in trace)

B Channel Configuration

Overview

This section describes the third step in the basic ISDN configuration sequence.

Requirement

The following is a requirement for configuring B channels:

- The D channel must be assigned.

B Channel Configure API message

Use the *B Channel Configure* message to configure B channels. Use care when configuring network-specific elements for the network. Check with your carrier if you need to include network-specific Information Elements.

Trunk-provisioned calls

If placing trunk-provisioned calls (options specific to a group of B channels), the host uses the parameters (configured with this message) in the outgoing ISDN SETUP message to the network. This data is used to encode the correct Information Elements for the called and calling party IEs when using trunk-provisioned parameters for the call type, bearer capability, numbering types, and plans.

Default Configuration

The table below lists the B channel configuration entities and their defaults. Use the *B Channel Configuration* message to change the default configuration.

Option	Default
Network Type	AT&T Megacom
Outgoing Information Transfer Capability	Voice
Calling Number Type	National Number
Calling Number Plan ID	ISDN Numbering Plan
Calling Presentation Indicator	Presentation Allowed
Calling Screening Indicator	User Provided, Not Screened
Called Number Type	Subscriber Number
Called Number Plan ID	Unknown Numbering Plan

Configuring the Backup D Channel

Purpose

This section describes an optional ISDN configuration for the backup D channel.

Before you begin

Download and assign custom PPL protocols and modify timers and configuration bytes as required.

D channel backup supported

ISDN cards support the D Channel backup procedure as defined by Lucent TR 41459. This procedure provides a standby D channel when using NFAS. When an active D channel fails, or if there is a span alarm, the CSP converts to the standby D channel and maintains active calls. This is a provisioned option and is applicable only when using NFAS. D Channel backup is supported on the T1 cards only.

Both D channels must be assigned to the same ISDN card. When a span fails, a D channel fails, or a T1 card is removed, a D channel switchover occurs (not a card switchover).

As both D channels must be assigned to the same ISDN card, there are no extra hardware requirements to enable D Channel Backup. You can logically assign spans on any T1 card in the system.

Configuring D channel backup

To configure the backup D Channel, perform the following steps:

Step	Action
1	Use the *D Channel Assign* message to configure the type and location of the D channel. After the primary D channel is configured, configure the backup D channel (using the *D Channel Assign* message again). Include the span, channel, and facility number on which it resides.
2	Send the *D Channel Facility List Configure* message to all other facilities you want to include in the NFAS arrangement. You can also send this message for the same facility that was assigned in the backup D channel assignment. The backup D channel cannot be de-assigned; it becomes de-assigned when the primary D channel is de-assigned.
3	Use the second facility (1) to configure the backup D channel; (however, 1-19 are permissible).
4	Bring spans and channels into service. It is not necessary to bring the backup D channel into service. It automatically comes into service when the primary D channel is brought into service.
5	When the D channels align and process the SERVICE/SERVICE ACK exchange, the host receives a *DS0 Status Change* message (indicating “In-Service”) for the primary D channel.

Call Processing

When one of the D channels becomes active, both incoming and outgoing call processing can begin. Only the active D channel allows information frames to progress to the Layer 3 Channel Released component 8 (where calls are managed). On a D channel switchover, only calls in the Active State (10) are preserved; all other calls are cleared. Messages are dropped in the Wait State when the switchover occurs. The host receives *Channel Released* messages for calls in transit.

Manual D Channel Switchover

You can initiate the switchover of D channels manually by sending a *PPL Event Request* message to the L3 D Channel Control component and the D channel to which you want to be switched.

Important! You cannot initiate a switchover by taking the primary D channel out-of-service or the span on which it resides.

Compliance

The CSP software is compliant with the Primary Rate Interface guidelines defined in Lucent TR 41459 specification, as follows:

Layer 2

Layer 2 provides information frames to Layer 3. For both D channels, all unnumbered, supervisor, and information frames are processed as described in the Lucent specification. Since the protocol is run over primary rate, the SAPI and TEI are both 0. Data links do not know if the D Channel Backup is enabled or disabled.

Layer 2 is managed by the following Layer 3 D Channel Control messages (primitives):

- DL Establishes Request
- DL Remove Request
- DL DM Remove Request
- DL Data Request

Layer 3

Layer 3 complies with Lucent specification TR41459. The L3 D Channel Control component (0x0A) handles the D channel backup logic. Each state represents both D channel states. The L3 DSM manages the following:

- Distribution of incoming and outgoing information frames to the other components in Layer 3

- The switchover procedure (sending SERVICE and SERVICE ACK messages on the D channel)
- Distribution of the indication to appropriate component to preserve active calls
- Reporting of D channel failure
- Reporting status to the L3P D Channel Control component (0x06)
- Reporting status to the host using the *PPL Event Indication* message

When a D channel changes state, the host receives one of the following events in the *PPL Event Indication* message:

- D is Active (0x00001)
- D is Standby (0x0002)
- D is not aligned (0x0003)

The D channel failure event is sent from L3 D Channel Control component to the L3P D Channel Control component only when both D channels are not aligned. L3P does not know the state of both D channels; it only knows that at least one D channel is accessible, or that both D channels are not.

Any call not in Active State gets cleared on a switchover. The host receives a *Channel Released* message for the bearer channel to which the call was in transition.

Layer 3 Plus

L3P automatically requests a switchover when either span goes into an alarm condition. The L3P B Channel Control component informs the host about channel service states. When both D channels are not established, it resends the *Alarm* message to the host to indicate the establishment of a D channel.

Optional Information Elements

Overview

Dialogic provides a format to send and receive Information Elements (IEs) per D channel. The data is represented using Information Control Blocks (ICB) with the ICB subtype of “Formatted IEs” (see *Information Control Blocks* in the *API Reference*). This allows flexibility when sending a message that has different types of data to transport. Multiple Formatted IEs can be loaded in the data section of the ICB.

APIs support IEs

The following API messages support the Formatted IE ICB:

- *Request For Service with Address Data*
- *Outseize Control*
- *PPL Event Indication*
- *LPPL Event Request*
- *Release with Data*
- *Channel Release Request*

Information Elements

The IE data is defined by the IE type. You can load multiple Formatted IEs in the data section of the ICB. Formatting specific IEs makes it easier for host development code to parse and create messages to pass specific data. The Raw IE ICB subtype is formatted similarly, however, it consists of the exact IEs to be sent.The format of the Formatted IEs ICB is shown in the table below.

ICB Count	0x01
ICB Type	0x02 (Data)
ICB Subtype	0x10 (Formatted IE)
ICB Data Length	User-defined
ICB Data[0] Number of IEs To Follow	User-defined
ICB Data[1] IE Type	User-defined
ICB Data[2] IE Length	User-defined
ICB Data[3] IE Data[0]	User-defined
ICB Data[4] IE Data[1]	User-defined

ICB Data[4] IE Data[2]	User-defined
ICB Data[4] IE DATA[3]	User-defined

Information Element Library

To implement overlay networks or proprietary applications, the IE library allows you to store commonly used IE data. The library loads specific ISDN messages during call processing to reduce host message traffic.

There is a separate IE library for each active D channel on the ISDN card. The IE library contains a maximum of 30 library entries, and each entry is a maximum of 30 bytes. Each IE library entry can hold multiple IEs including codeset 6 and 7 IEs. IEs can be any type.

IE Restrictions

The following restrictions apply to the contents of an IE library entry

- The IE lengths must coincide with total length of data to be stored.
- The total length of the entry must be 30 bytes or less.
- The length specified internally for the IEs must be consistent.

Configuration

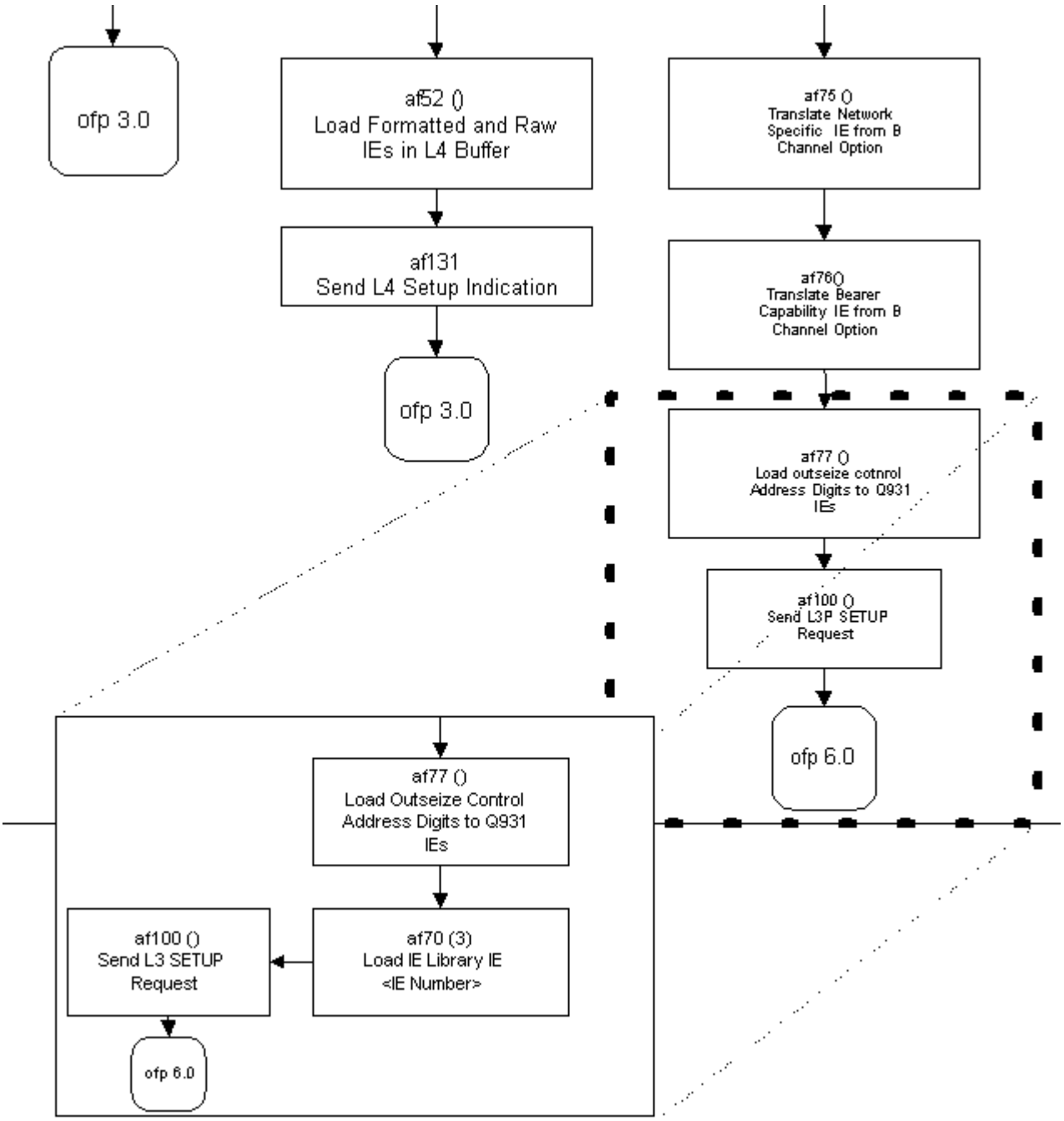
Use PPL AF 70 in the L3P Call Reference component to insert an IE Library into an outgoing ISDN message. The first argument is the IE library entry number to insert in the ISDN message.

Figure 8-6 shows page 1 of the L3P Call Reference State Machine modified to use AF 70 to insert the IE into the outgoing SETUP message to the network. The first argument is the IE Library entry number (3). in the *ISDN Interface Configure* message example.

Use the *ISDN Query* message to retrieve the contents of the IE library. The IE library is stored in battery backed RAM.

Figure 8-6 shows page 1 of the L3P Call Reference State Machine (l3p_cc). In the modified insert, AF 70 has been added after AF 77, with Argument 1 indicating IE Number 3. This inserts the IE into the outgoing SETUP message to the network.

Figure 8-6 L3P Call Reference DSD - Page 1



Examples

The following example shows an *ISDN Interface Configure* message used to load an IE library entry. Use the *ISDN Query* message to retrieve the contents of the IE library.

Byte	Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x0020 (32)
3, 4	Message Type	0x00B2
5	Reserved	0x00
6	Sequence Number	0xSN
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Address Data Length	0x03
12, 13	Address Data[0,1] Logical Span	0x0001 (Span 1)
14	Address Data[2] Channel	0x17 (Channel 23)
15	Entity	0x0B (Load IE Library Entry)
16	Data[0] Entry Number	0x04
17	Data[1] IE Type	0x01 (Q.931 IE)
18	Data[2] Total IE Length	0x0E (14)
19	Data[3] IE Data	0x70 (Called Party Number IE)
20	Data[4] IE Data	0x0C (Length of Called Party)
21	Data[5] IE Data	0xA1 (National number, ISDN Numbering)
22	Data[6] IE Data	0x31 (Digit 1)
23	Data[7] IE Data	0x35 (Digit 5)

Byte	Description	Value and Indication
24	Data[8] IE Data	0x30 (Digit 0)
25	Data[9] IE Data	0x38 (Digit 8)
26	Data[10] IE Data	0x38 (Digit 8)
27	Data[11] IE Data	0x36 (Digit 6)
29	Data[12] IE Data	0x32 (Digit 2)
30	Data[13] IE Data	0x33 (Digit 3)
31	Data[14] IE Data	0x30 (Digit 0)
32	Data[15] IE Data	0x30 (Digit 0)
34	Data[16] IE Data	0x30 (Digit 0)
35	Checksum	0xCS (not shown in trace)

Other Optional ISDN Configurations

Overview

Besides D channel backup and Information Elements, there are other optional configurations that are available with ISDN PRI.

This section outlines the configurations for:

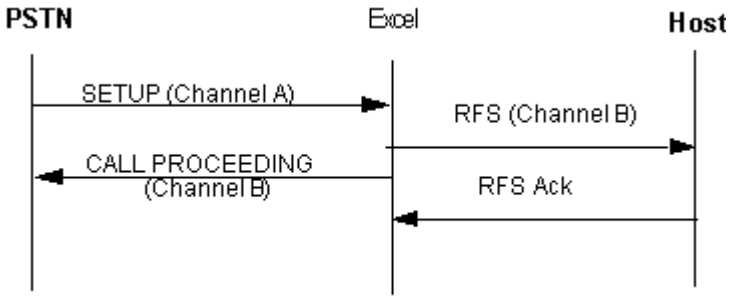
- B channel negotiation
- ANI on demand
- Vari-a-bill billing

B Channel Negotiation

ISDN cards support B channel negotiation on incoming calls as defined in Lucent TR41459 for both E1 and T1 cards. This feature allows the user receiving the incoming call (SETUP) to negotiate for the selection of the B channel. Only B channels controlled by the same D channel are considered for the selection procedure.

Call Flow

The SETUP message indicates the channel as either exclusive (does not accept an alternative) or preferred (any alternative is acceptable). If the channel is specified as preferred, but the user cannot grant the indicated channel, it selects any other B channel associated with the D channel. If the channel was specified as exclusive, no negotiation of the selection of the B channel is permitted and the call is rejected.



ISDN Interface Configure enables feature

B channel negotiation is enabled using the *ISDN Interface Configure* message. The following selection modes are provided:

- Linear Clockwise (default)
- Linear Counter-Clockwise

- Circular Clockwise
- Circular Counter-Clockwise

For NFAS, B channels on the same facility are selected before channels on other facilities.

Sample usage of feature

The following example shows the use of the *ISDN interface Configure* message to select the B channel negotiation mode.

X->H FE [00 0F] [00 60] 00 00 01 00 01 0D 03 00 00 17 06 00 01]

Byte	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x000F (15)
3, 4	Message Type	0x0060
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Address Data Length	0x03
12, 13	Address Data[0] Logical Span	0x0000 (Span 0)
14	Address Data[1] Channel	0x17 (Channel 23)
15	Entity	0x06 (B Channel Selection Mode)
16	Data[0] Reserved	0x00
17	Data[1]	0x01 (Linear Clockwise)
18	Checksum	0xCS (not shown in trace)

ANI on Demand

ISDN cards support the Automatic Number Identification (ANI) on Demand feature as specified in Lucent TR41459. This feature allows the host to request the Calling Party Number from networks that support ANI such as AT&T MultiQuest.

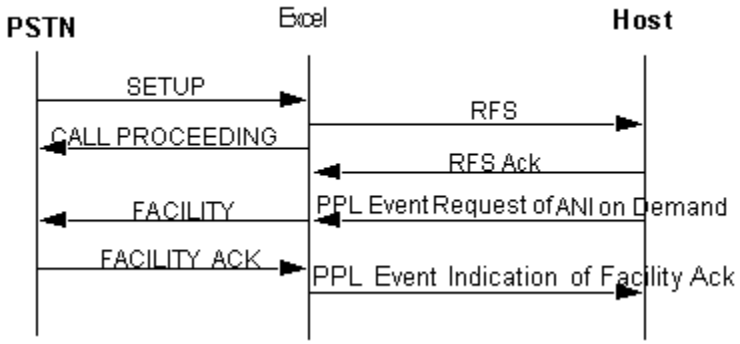
A *PPL Event Request* message of ANI on Demand (0x21) must be sent from the host immediately after receiving the *Request For Service* message from the CSP. If the call has already been answered, a NACK of 0xC0 is returned by the CSP.

If successful, a FACILITY message is sent to the network requesting the ANI. A *PPL Event Indication* message of FACILITY ACKNOWLEDGE is returned containing the ANI information in a Calling Party IE or Network-specific Facility IE. If the ANI is not supported by the network or is not available, a *PPL Event Indication* message of FACILITY REJECT is returned by the CSP.

See the *Lucent TR41459* specification for detailed information.

ANI on Demand Call flow

The figure below shows a call flow using the *PPL Event Request* message to implement the ANI On Demand feature.



Vari-A-Bill Billing

ISDN cards support the Vari-A-Bill feature as specified in Lucent TR41459. This feature allows the host to request a change in billing for an incoming call. You can use this feature after a call has been answered and only if the SETUP message indicates that the network supports the flexible billing service.

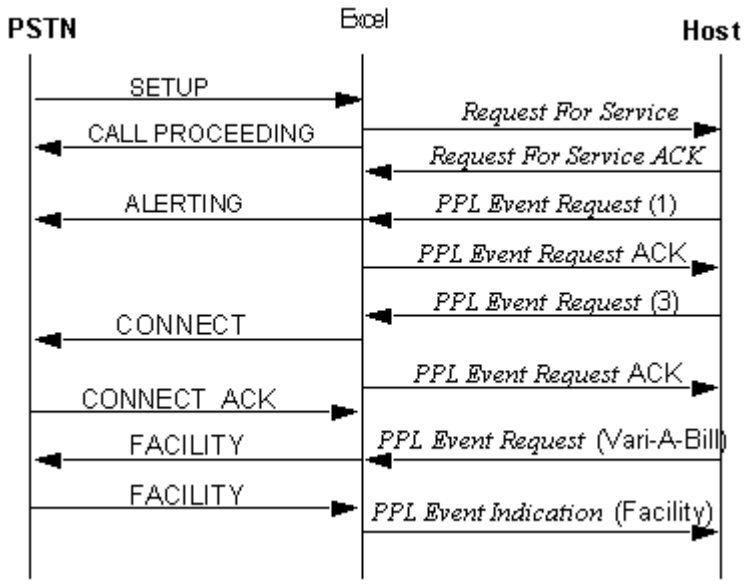
Enabling Vari-a-Bill

To enable Vari-a-Bill, send a *PPL Event Request* message with a PPL Event of Vari-A-Bill (0x20), specifying the new billing information in a Vari-A-Bill Data ICB (0x12). The CSP sends a FACILITY message to the network requesting the billing change.

If the call is not in the answered state, then the feature is not supported and a NACK of 0xC0 is returned by the CSP. A *PPL Event Indication* of FACILITY is returned from the network indicating whether the billing change was successful. See the *Lucent TR41459* specification for detailed information.

Call Flow

The following call flow shows the use of the *PPL Event Request* message to implement the Vari-A-Bill feature.



Example

The following example shows the *PPL Event Request* message to implement Vari-A-Bill billing and change the billing rate.

Trace X->H FE [00 1E] [00 44] 00 00 01 00 01 0D 03 00 00 00 [00 05] 00 20 01 02 10 0A 01 12 07 01 90 30 31 32 35 36

BYTE	Field Description	Value and Indication
0	Frame	0xFE
1, 2	Length	0x001E
3, 4	Message Type	0x0044
5	Reserved	0x00
6	Sequence Number	0x00
7	Logical Node ID	0x01
8	AIB Address Method	0x00 (Single Entity)
9	Number of Address Elements	0x01
10	Address Element Type	0x0D (Channel)
11	Address Data Length	0x03
12, 13	Address Data[0] Logical Span	0x0000 (Span 0)
14	Address Data[1] Channel	0x00 (Channel 0)
15, 16	PPL Component	0x0005 (L3P Call Reference)
17, 18	PPL Event	0x0020 (Vari-A-Bill)
19	ICB Count	0x01
20	ICB Type	0x02 (Data)
21	ICB Subtype	0x10 (Formatted IE)
22	ICB Data Length	0x0A
23	ICB Data[0]Number of IEs To Follow	0x01
24	ICB Data[1] IE Type	0x12 (Vari-a-bill Data)
25	ICB Data[2] IE Length	0x07
26	ICB Data[3] Invoke ID	0x01
27	ICB Data[4] Billing Type	0x90 (New Rate)

BYTE	Field Description	Value and Indication
28	ICB Data[5] Billing Data (Hundreds)	0x30
29	ICB Data[6] Billing Data (Tens)	0x31
30	ICB Data[7] Billing Data (Ones)	0x32
31	ICB Data[8] Billing Data (Tenths)	0x35
32	ICB Data[9] Billing Data (Hundredths)	0x36
33	Checksum	0xCS (not shown in trace)

ISDN Call Processing

Overview

Understanding the various phases of the call makes it easier for you to manage the events and information flow. This section describes the call phases and events and identifies what you must do to use the call processing capabilities. This section also describes the level of control for call processing that you must engineer. The topics in this section include:

- Incoming Calls
- Outgoing Calls
- Releasing Calls
- Overlap Receiving
- Overlap Sending

Before you begin

Before you continue, make sure the following have occurred:

- System configuration is complete
- Spans are framed
- All D channels are established
- All B channels are in service

The *Outseize Instruction List Configure* message is not supported for ISDN PRI. All outseize instructions must be sent in the *Outseize Control* message.

Perform all PPL configuration after you specify the variant with the *ISDN Interface Configure* message.

Incoming Calls

An incoming call is indicated by a SETUP message containing IEs to describe the span, channel, bearer capability, called and calling party, and services requested. The CSP validates and verifies the information based upon the ISDN PRI variant and, if successful, returns a CALL PROCEEDING or SETUP ACK. A SETUP ACK returns if the ISDN PRI variant is set to Euro-ISDN (0x07) or Euro-ISDN Netside (0x17), and the overlap receiving mode is enabled. Otherwise, only the CALL PROCEEDING returns from the CSP.

Normally, only the span, channel, called and calling party are needed for most applications. This is provided in the *Request for Service with Data* message using the default BCD Encoded Stages format. This is the same message, common across the CSP, that reports the called and calling party address digits.

> **Important!** The *Inseize Control* and *Inseize Instruction List Configure* messages are not supported for ISDN PRI.

If access to all IEs is required, configure the Request for Service Format to Formatted ICB. Formatted IEs reformats the most used IEs to a byte-oriented format which is easier to parse. This abstracts the ISDN specifics from the host to provide what is necessary, in a generic form. IEs that are not supported in the formatted mode are sent to the host in the Raw IE ICB.

The Request For Service Format is configured with Config Byte 1 of the L3P Call Control PPL Component (0x05). The following options are configurable:

- BCD Encoded Digits (0x01)
- Formatted ICB (0x03)
- Exact Frame (0x04) - (host must parse)

At this point in the call, the network is waiting for a CONNECT message. Prior to this, an ALERTING or PROGRESS message informs the network that the CSP is awaiting answer on the B party of the CSP. One method of sending the ALERTING message to the network is to send a *Park Channel* message for the incoming span/channel. This disables Layer 4 timers for the call.

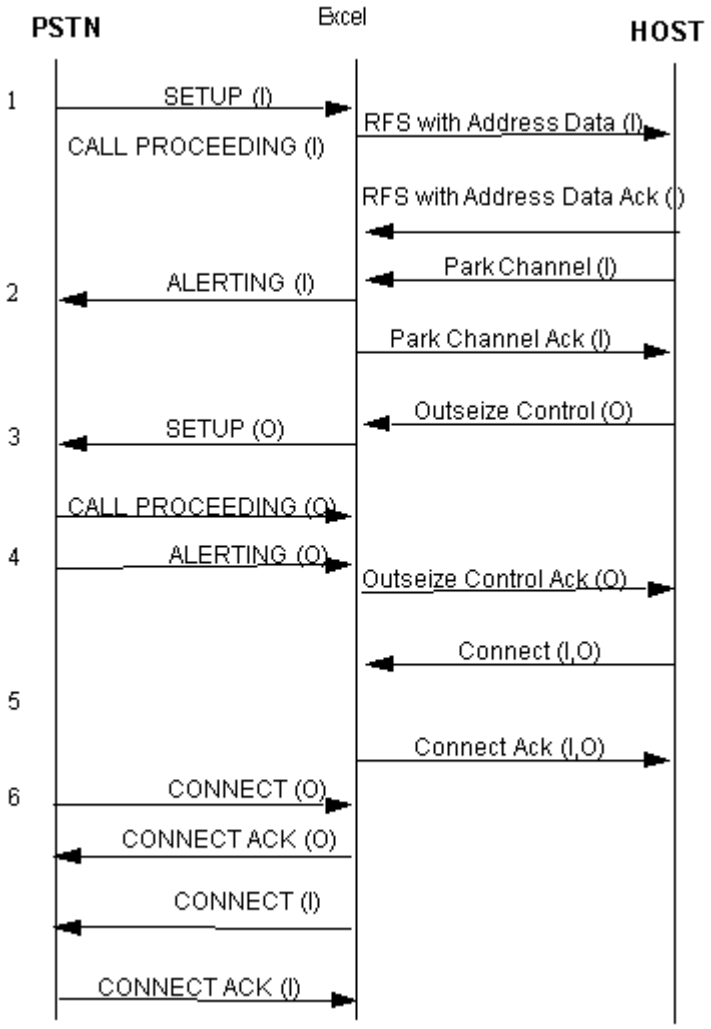
Normally, the host attempts an outgoing call (*Outseize Control*) to connect to this incoming call. If a *Connect* message is sent to the CSP after a successful seizure, the answer propagates to the incoming side (CONNECT) assuming the *Answer Supervision Mode Configure* message is set to Propagate Answer To Distant End (default).

To automatically answer an incoming call, send the *Generate Call Processing Event* of answer, which sends an ALERTING followed by a CONNECT. This puts the call in an answered state and incoming billing is started.

Call Flows for Incoming Calls

Incoming ISDN Call to Outgoing ISDN Call

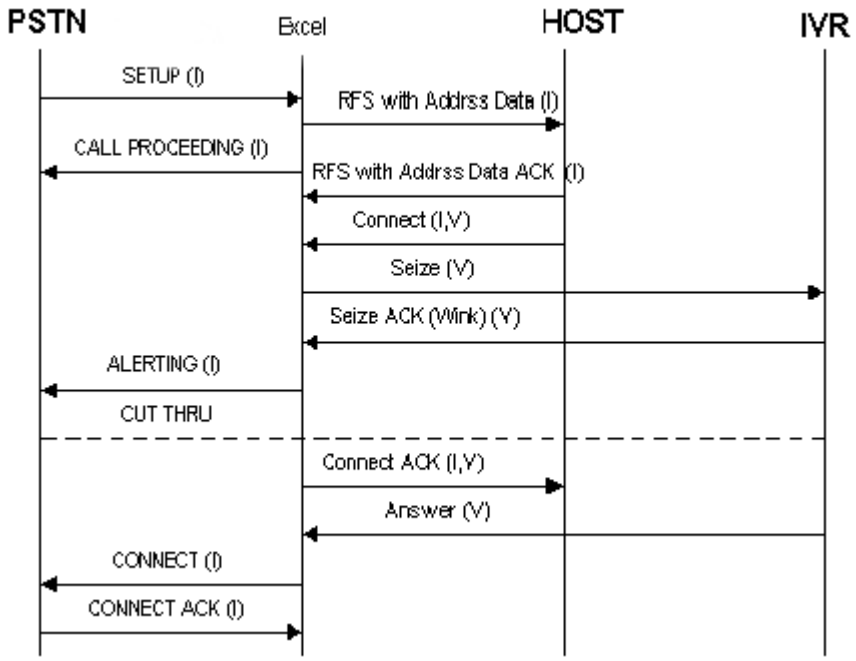
The following call flow shows an incoming ISDN call (I) from the network. For this application, another ISDN call (O) is generated on a different span/channel combination.



1. Upon receiving the SETUP from the CO, the IEs are validated and the CALL PROCEEDING is automatically returned. The SETUP is transformed into the *Request for Service with Address Data* and sent to the host.
2. The host parks the channel, which causes the ALERTING message to be sent for that call.
3. The *Outseize Control* message is sent to originate a call. For the common API approach the instructions should be:
   - Seize
   - Send Host ACK
   - Wait for Host Control
   - Data: Stage 1 Digits
4. Upon the reception of the ALERTING or PROGRESS, the *Outseize Control* ACK is returned to the host.
5. To establish a 2-way circuit switched connection between the channels, the host issues a *Connect* message.
6. Once the outgoing call is answered, the connect is propagated to the incoming side only if the answer supervision is configured to be “propagated.” See the *Answer Supervision Configure* message.

Incoming ISDN Call to an IVR (Interactive Voice Response)

The following call flow shows an incoming ISDN call with a seize and connection to an IVR port with a trunk type of E&M Wink Start.



1. The host sends a *Connect* or *Connect with Data* message to force the second span/channel combination to seize and then gets connected to the incoming call. Use this message to seize the second party if the trunk type does not require digit outpulsing. ISDN outgoing calls must have the Called Party specified in the *Outseize Control* message. The connect messages cannot be used to originate an ISDN call.
2. Assuming that the IVR is configured to the trunk type E&M Wink Start when the wink has returned the ALERTING message is propagated to the incoming side and has the Information Elements (IE), if any, passed in the *Connect with Data* message. Detecting the outseize ACK (wink) also causes the circuit switched connection to occur.
3. Sometime later the IVR answers, which propagates the CONNECT to the ISDN.

Incoming ISDN Call Answered and Parked

The host can send a *Generate Call Processing Event* of “answer” to generate the appropriate ISDN signaling to the network. The CSP will park the call in this scenario and the incoming caller will hear silence until reconnected or tones/announcements are played.

Sending Outgoing Calls

To send a Setup to the network, use the *Outseize Control* message. The easiest method to use for inserting called and calling party digits is to use the Stage N Address Data ICB. All other necessary ISDN specific information derives from the configured options of the *B Channel Configure* message. Information, such as number types and plans, is preconfigured using this message.

For call-by-call service selection, send the *Outseize Control* message using the ISDN Formatted IEs and ISDN Raw IEs ICBs, which is the recommended method. When doing this, specify the number type and plan for the called party IE and the calling party IE. You can also specify other IEs to be inserted.

Several IEs are inserted in the outgoing SETUP automatically:

- The ISDN header (protocol discriminator, call reference and message type)
- The Channel Identification IE
- Bearer Capability IE
- Any IEs configured and implemented in the PPL

The B channel configuration options are interpreted and inserted automatically if not represented in the *Outseize Control* message (see the section on *B Channel Configuration (8-37)* for the table). Inserting IEs in the *Outseize Control* message override the configured values and are inserted in the SETUP.

Once the SETUP is sent, the CSP waits for a CALL PROCEEDING ACK, or a SETUP ACK. Once the ALERTING, PROGRESS, or CONNECT occurs, the *Outseize Control* ACK returns to the host. You can send each of these events to the host with the data included using the *PPL Event Indication* message. Once the ACK is sent, the host performs any of the Layer 4 specific call processing events.

When the CSP receives no response from the network

If the CSP does not receive a network response to the SETUP in eight seconds, it responds to the host *Outseize Control* message with a state of Outseize Failed, No Answer (0x1B). Eight seconds is two full cycles

of the Q.931 Timer 305, which is L3 Call Control PPL Timer 5 and has a default value of four seconds. The SETUP is rejected by L3 if any IEs are invalid or missing.

The CSP responds to the host *Outseize Control* message with a status of 0x17 (Invalid Data Type). The L3P Call Control PPL sends an ACK to the host for an *Outseize Control* message. The default PPL does not examine the Send Host ACK ICB (0x08) in the *Outseize Control* message.

If involved in a tandem connection, reception of the ALERTING, PROGRESS, or CONNECT causes the voice path to get cut through. This is controlled by changing the PPLs.

In the case of distant-end release (as a result of the *Outseize Control*), if the called party is busy, a DISCONNECT or RELEASE COMPLETE is sent to the CSP and a *PPL Event Indication* is sent to the host with the Cause IE indicating the reason.

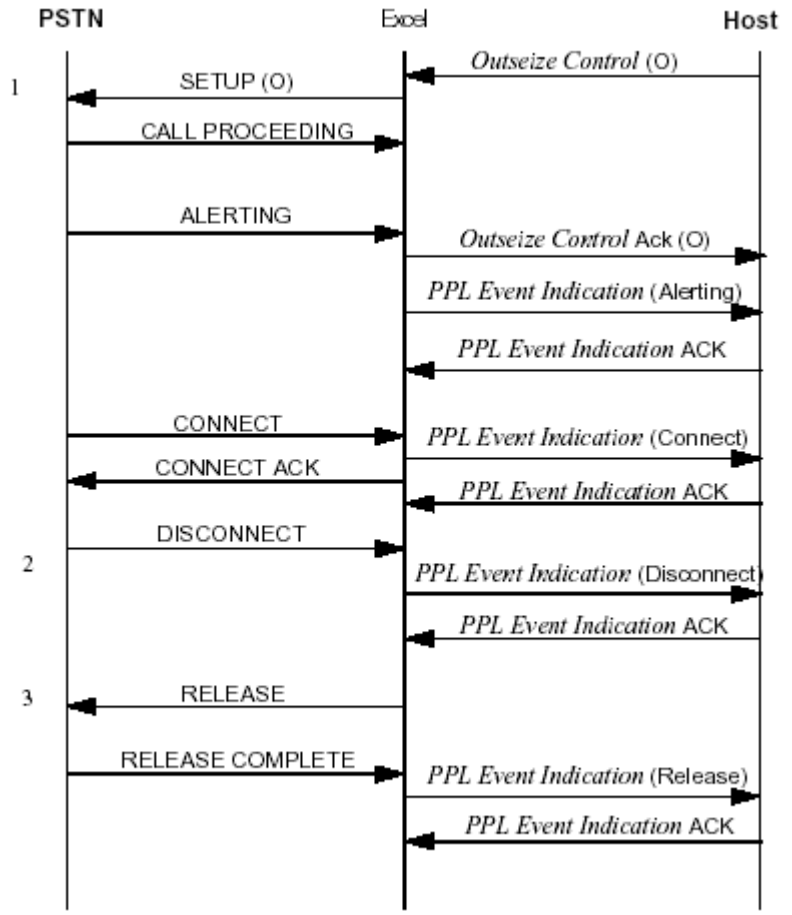
Important! Outseize options are configurable by modifying the PPL configuration bytes of the L3P Call Control component (ID 0x05). You must modify the config bytes to enable the following options:

- Config. Byte 11 (Euro ISDN/Austel only): Send a SETUP ACK and *PPL Event Indication* of More Info to the host upon receipt of SETUP ACK from the network.
- Config. Byte 12: Send a *PPL Event Indication* of Call Proceeding to the host upon receipt of a CALL PROCEEDING indication from the network.
- Config. Byte 13 (Euro ISDN/Austel only): Send an *Outseize Control* ACK to the host upon receipt of a SETUP ACK from the network.
- Config. Byte 14: Send an *Outseize Control* ACK to the host upon receipt of a CALL PROCEEDING message from the network.

Call Flows for Outgoing Calls

Outgoing ISDN Call with Raw Formatted Data Call Processing Events

The following call flow shows an outgoing ISDN call with the *PPL Event Indication* bytes set to report each event. Upon reception of one of these messages, if configured it gets reported to the host. This assumes that the *Outseize Control* message has the following instructions:

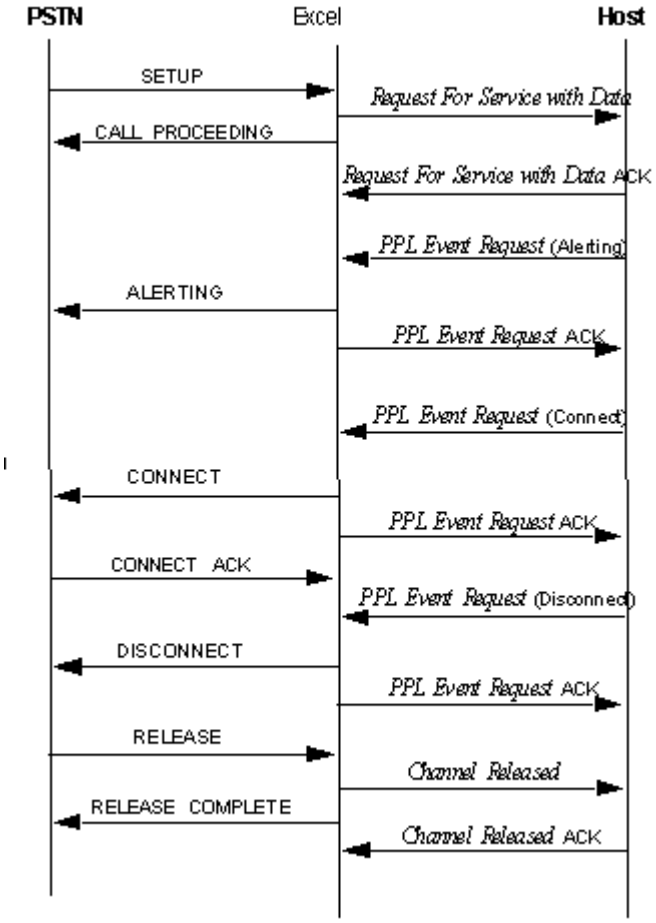


1. Seize
2. Send Host ACK
3. Wait for Host control
4. Data: Stage 1 Digits
5. The Host ACK is sent as soon as the CSP validates the *Outseize Control*.
6. All of the ISDN messages after the SETUP can be sent to the host in a *PPL Event Indication* message. The Information Elements (IE) are parsed and certain IEs are put into the message to the host depending on the message type (see *PPL Event Indication 0x43* in the *API Reference*).

The DISCONNECT IEs could be reported to the host, as shown here in a *PPL Event Indication* message.

Incoming ISDN Call with Host Control of Call Setup and Release

The following call flow and all optional information elements to be included with each ISDN message. You can include optional IEs with any of the *PPL Event Request* host messages. These IEs are validated and then issued with the ISDN message. The ISDN PPLs ensure that the mandatory IEs are included with each ISDN message.



Releasing Calls

You can release a call in the following ways:

Network-Initiated

For a network-initiated release, a DISCONNECT is received. If the appropriate PPL configuration byte is set, release is reported to the host in a *PPL Event Indication* message. It could be parsed for access to the IEs by the application. By default, the CSP returns the RELEASE message and once the RELEASE COMPLETE is received the CSP sends the *Channel Released* message to the host.

Release Channel or Release with Data messages

For host-initiated release, send either the *Release Channel* or the *Release with Data* message. If not specified, the Normal Clearing (0x10) cause code is inserted. Use the *Release with Data* message to insert IEs if needed.

PPL Event Request message

Send a *PPL Event Request* message with an event of DISCONNECT to the L3P Call Control component (0x05).

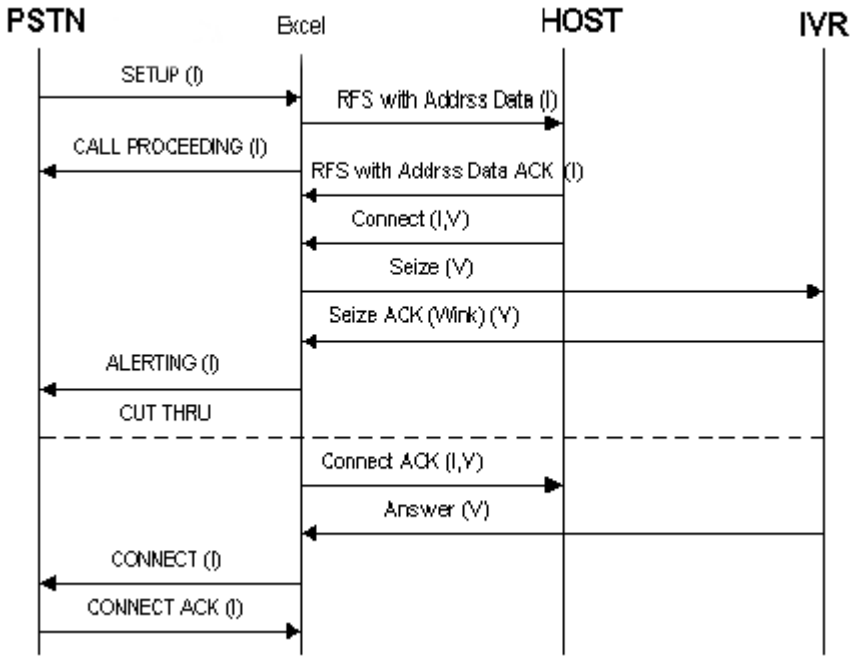
Channel Release Request message

A more flexible method of release is to send the *Channel Release Request* message followed by a *Release with Data* message, which sends the RELEASE out with application-selected information elements.

When the distant-end releases in a tandem connection, the Channel Release Request message is sent to the host to tell it to send the Release with Data message to specify IEs that go into the DISCONNECT message. For backward-compatibility with the ISDN PRI-24 implementation, this message is in the current software; however, the best approach is to change the PPL. The *Channel Release Request* and *Release with Data* messages only support the ISDN Formatted IEs and ISDN Raw IEs ICBs for ISDN.

ISDN-initiated Release after Connection to an IVR (Interactive Voice Response)

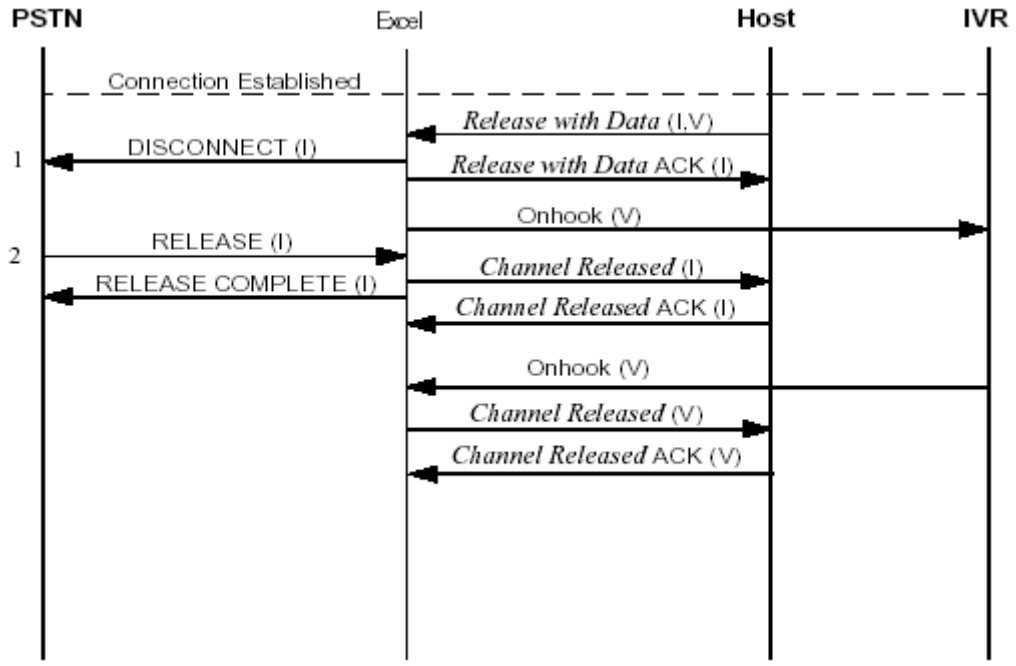
This call flow shows an existing ISDN to E&M Wink Start interface to an IVR and the ISDN initiating normal call clearing procedures.



1. If the host had the option of “Channel Release Request on ISDN Disconnect” bit set in the *ISDN Interface Configure* message, the *Channel Release Request* message would get sent to the host. Otherwise, the host would not be notified of the call being cleared until receiving the *Channel Released* message. Also, as shown above, if the host did not have “Distant End Release Mode” or “Local End Release Mode” set to park, the CSP would automatically initiate clearing procedures on the channel included in the connection (IVR).
2. The host could insert specific Information Elements in the RELEASE message by using the *Release with Data* message.
3. The on-hook from the distant end (IVR) could occur at any time and would get reported to the host using the *Channel Released* message.
4. When detecting the RELEASE COMPLETE, if the host does not need to know the Information Elements then only the *Channel Released* message is reported. Otherwise, use the *Call Processing Event* message.

Host-initiated Release after ISDN Connection to an IVR (Interactive Voice Response)

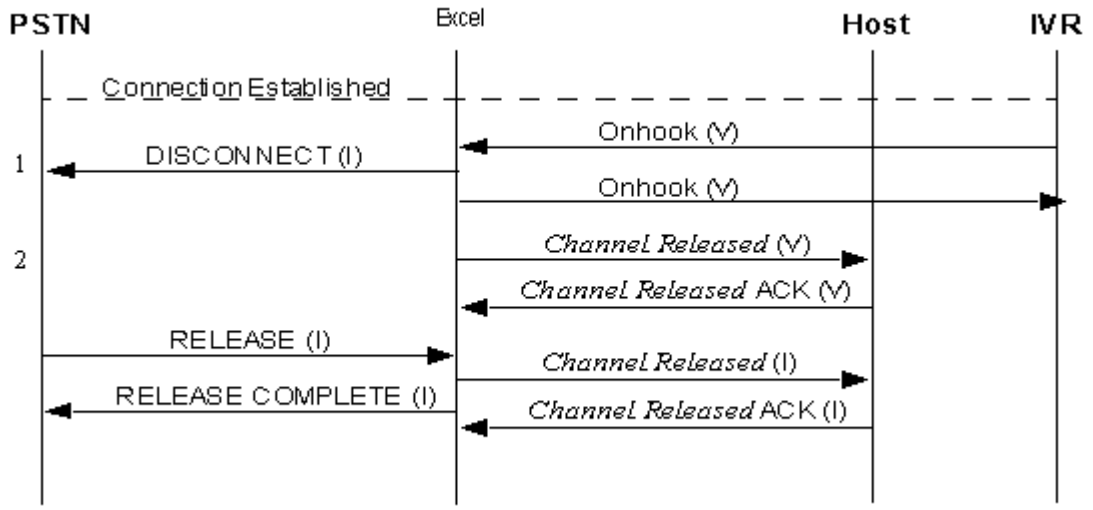
The following call flow shows an existing ISDN to E&M Wink Start interface to an IVR, and the host initiating normal call clearing procedures on the ISDN call.



1. If the host needs to insert its own Information Elements in the DISCONNECT, then send the *Release with Data* message. Specify both parties in the connection or the other channel (IVR) gets parked.
2. Once the CSP receives the RELEASE message, the RELEASE COMPLETE is returned and a *Channel Released* message is reported to the host.

IVR-initiated Release after ISDN Connection

The following call flow shows an existing ISDN to E&M Wink Start interface to an IVR, with the IVR initiating normal call clearing procedures for the connection.



1. If distant end release mode and local end release mode are not configured to park, then the connection automatically gets cleared and a “normal clearing” cause is sent to the ISDN.
2. Both channels send a *Channel Released* message to the host.

Overlap Receiving

Overlap receiving allows you to provide more information for a call if the SETUP does not contain enough information to route the call.

To use overlap receiving for incoming calls, select the Euro-ISDN PRI variant (0x07) or Euro-ISDN Netside PRI variant (0x17) using the *ISDN Interface Configure* message. Also, Overlap Receiving must be enabled by using the *PPL Configure* message to set PPL configuration byte 9 of the L3P Call Control component (0x05) to a value of 0x01. Once the CSP is configured for overlap receiving, the call setup process is transparent and appears to the host as a normal call setup.

When overlap receiving is enabled, the CSP validates the IE data in the incoming SETUP message as a usual call setup. The CSP also checks for the following:

- A sufficient number of digits in the Called Party IE. This number is determined by configuration byte 10 of the L3P Call Control component (0x05), which defaults to a value of 0x10.
- A Sending Complete IE in the SETUP message

If both conditions are false, the CSP sends a SETUP ACK message to the network and goes into overlap receiving mode to obtain more information. If one of the conditions is true, the CSP sends a CALL PROCEEDING and continues with normal call setup.

In overlap receiving mode, the CSP waits for one or more INFORMATION messages from the network. These INFORMATION messages contain the Called Party digit data. Overlap receiving continues until one of the following conditions occur:

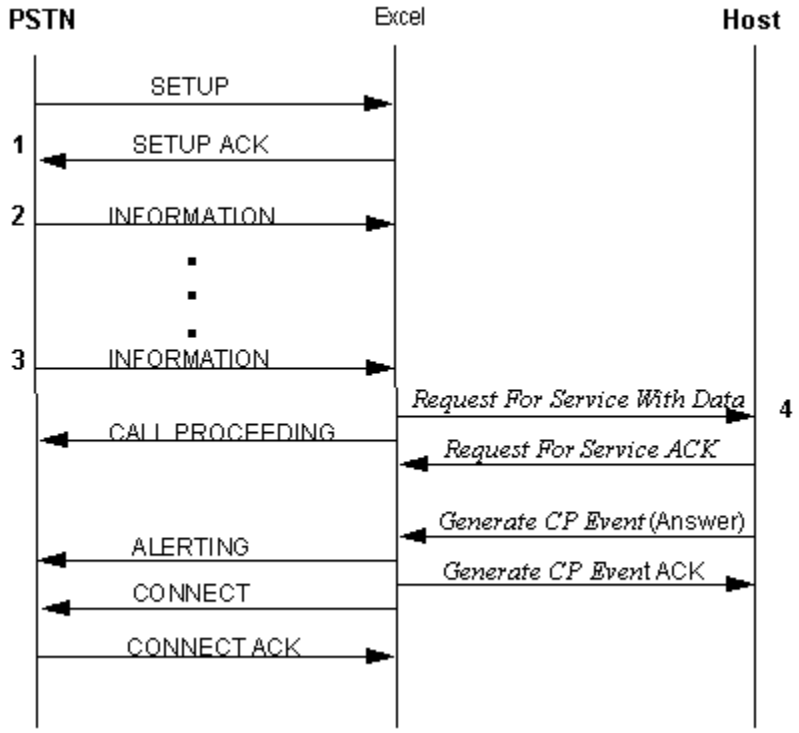
- The total number of Called Party digits that the SETUP receives and all the subsequent INFORMATION messages are a sufficient number
- An INFORMATION message receives a Sending Complete IE
- A timeout condition occurs

In the first two conditions, the CSP sends a CALL PROCEEDING message to the network and a *Request For Service With Data* message is sent to the host with all of the Called Party digits. When a timeout condition occurs, a 15 second L3 Q.931 timer (T302) is set after the CSP sends a SETUP ACK. This timer is reset when each INFORMATION message is received. If this timer expires before the CSP receives an INFORMATION message, the incoming call is cleared with a DISCONNECT message.

As an option, you can override the L3 timer to allow all calls to complete. To do this, configure timer number 2 (using the *PPL Timer Configure* message) in the L3P Call Control component (0x05) to a value of less than 15 seconds.

If the L3P timer expires before the network receives an INFORMATION message, the CSP sends a CALL PROCEEDING message to the network and a *Request For Service With Data* message, with all of the Called Party digit data collected up to that point, to the host. You can also use the L3P timer when the sufficient number of digits is indeterminate.

The following call flow shows an incoming call from the network using overlap receiving.



1. If overlap receiving is enabled, the CSP responds to the SETUP with a SETUP ACK.
2. INFORMATION messages received contain additional Called Party digits.
3. The final INFORMATION message may contain a Sending Complete Indication (sending complete IE or # in Called Party IE) or, at this point, the total number of called party digits meets the sufficient number of digits (specified in the value of configuration byte #9 for component #5).
4. The *Request for Service* message is sent when enough digits are collected and then CALL PROCEEDING is sent.

Overlap Sending

When you select the Euro-ISDN PRI variant (0x07) using the *ISDN Interface Configure* message, you can use overlap sending for outgoing calls. To select this option, include a Call Type ICB (0x1A) in the *Outseize Control* message. This ICB tells the CSP to use overlap sending and also determines which mode of overlap sending to use. The CSP uses two methods of overlap sending corresponding to Call Types 1 and 2.

Each Call Type is specified in the first data byte of the Call Type ICB:

- Call Type 1 (L3P-Controlled) -- Include all Called Party digit data that is not in the SETUP in the Call Type ICB as ASCII/IA5 digits following the Call Type byte. When the host sends an *Outseize Control* message, the CSP indicates a SETUP with whatever Called Party data was included in the Address Data ICB (ASCII/IA5 or ISDN Formatted IEs).

  Once the CSP receives a SETUP ACK message from the network, it sends a barrage of INFORMATION messages, one for each of the Called Party digits included in the Call Type ICB. The last INFORMATION message contains a Sending Complete IE. Use this mode mainly for testing purposes.

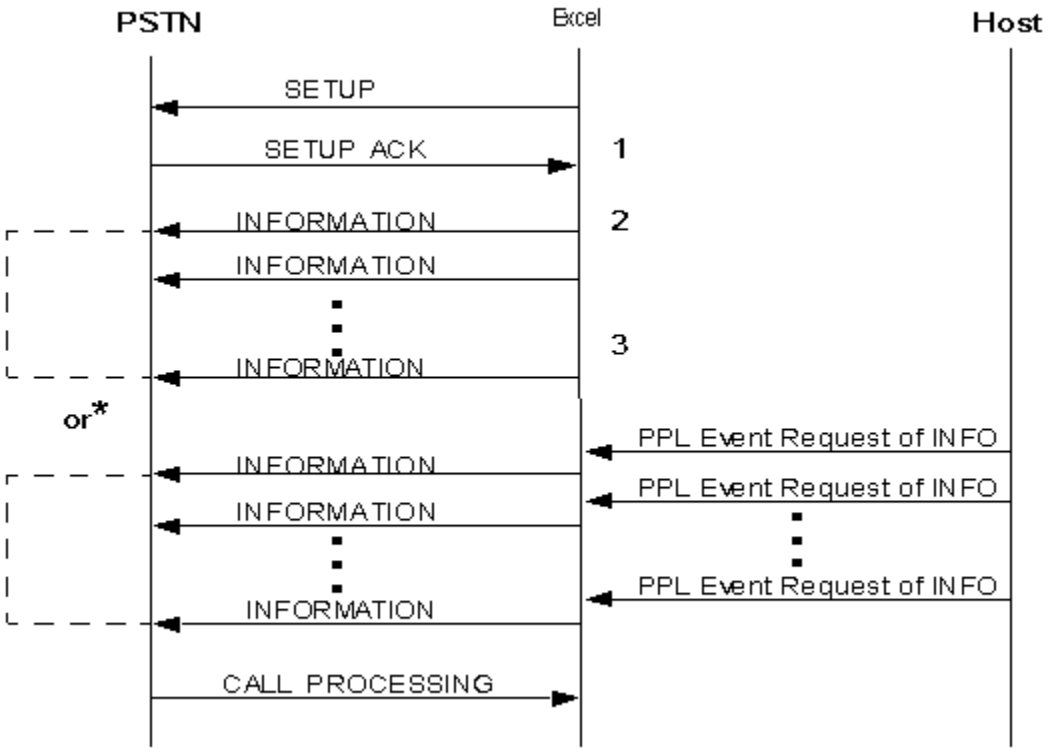
  - Call Type 2 (Host-Controlled) – The Call Type ICB should contain only the Call Type byte 2 as ICB data. In this mode, the CSP sends a SETUP message and waits for a *PPL Event Request* message from the host to initiate an outgoing INFORMATION message. Send a *PPL Event Request* message with an event of INFORMATION (0x0D) to the L3P Call Control component (0x05).

  The ICB data should be a subtype of ISDN Formatted IEs (0x10) and must contain a Called Party IE with one or more digits. The host can also include a Sending Complete IE as an ISDN Raw IE ICB if this is the last INFORMATION message to be sent.

As with overlap receiving, there is a Layer 3 Q.931 timer running (T304). It is a 30 second timer and is set when a SETUP ACK is received. It is reset when the host initiates an INFORMATION message using a *PPL Event Request* message. It is disabled when the CSP receives a message from the network that terminates overlap receiving, for example, a CALL PROCEEDING, ALERTING, or CONNECT. When the timer expires, the CSP clears the call by sending a DISCONNECT to the network.

To override this timer, set the L3P Call Control component (0x05) timer number 3 to a value of less than 30 seconds. When the L3P timer expires, the CSP sends an INFORMATION message to the network with a Sending Complete IE to terminate overlap sending and to continue call setup.

The following call flow shows an outgoing call to the network using overlap sending:



* The overlap sending is either host-controlled or L3P-controlled depending on which Call Type ICB is specified in the Outseize Control message.

1. Once SETUP ACK is received, the CSP sends INFORMATION messages that contain the remaining Called Party digits.
2. INFORMATION messages are sent from L3P.
3. INFORMATION messages are sent from the host using the *PPL Event Request*.

ISDN Redundancy

Overview

The ISDN PRI Redundant I/O card used with two ISDN PRI cards, or two ISDN Series 3 cards used with two CCS Series 3 I/O cards, provide complete redundancy by enabling all 32 D channels on a card to be replicated on a standby card. You configure redundancy by using the *CCS Redundancy Configure* message to designate one ISDN card as primary (active) and the other as secondary (standby) prior to configuring the D channels.

A mirror image of the active card's primary and secondary D channels is copied to the standby card. If the active card fails or you remove it, the standby card takes over and manages call processing. All calls in a connected state are retained while others are purged. If you reset the active card, it switches over to standby.

Configuration

To configure card redundancy, you must designate one ISDN card as primary and the other as secondary. The primary card attempts to become active. You refer to the primary slot number for all subsequent configuration.

CAUTION

Once redundancy is configured and the primary card becomes active, you should refer to the primary slot number only in subsequent configuration messages (D Channel Assign, PPL Audit Configure, PPL Audit Query).

You designate primary and secondary cards by sending the *CCS Redundancy Configure* message prior to assigning the D channels and completing the rest of a normal configuration.

The prerequisites for using this message are:

- Both ISDN cards (primary and secondary) must be in adjacent slots.
- Both ISDN cards must be in service.
- The ISDN Redundant I/O card must be installed in the appropriate slots, opposite the ISDN cards, in the back of the chassis.
- For each ISDN Series 3 card, you must use a single slot CCS I/O Series 3 card. The two adjacent CCS I/O Series 3 cards are linked with a CCS I/O Series 3 Redundancy Link cable.

After you send the message and a positive acknowledge (ACK) response is returned to the host, the primary card is placed in active mode and the secondary is placed in standby mode. All database information on the active card, including configuration data and active calls, is copied to the standby card. This process enables both the active and standby cards to manage the same information and input.

Important! Dialogic recommends that you send the *CCS Redundancy Configure* message after resetting the configuration on the whole system. The synchronization of databases on the cards involves intensive processing, so you should send the message when there is little or no call activity on the system. After you send the *CCS Redundancy Configure* message, all subsequent configuration is distributed to both cards.

Switchover

The CSP transfers control from the active card to the standby card when any of the following occurs:

- You remove the active card.
- You reset the active card by pushing the reset button.
- The active card malfunctions and forces a reset.
- The active card stops communicating with the Matrix Controller.
- The active card is taken out of service from the host.
- The host sends a *Reset Configuration* message to the active card.

During an ISDN card switchover, the stable calls remain active and the other calls are purged with the switchover purge code 0x18.

When the standby card becomes active, it takes over the management of the D channels. The host is informed by the *CCS Redundancy Report* messages and is then responsible for reconfiguring redundancy.

The host uses *CCS Redundancy Query* messages to track the state of both the primary and secondary cards. The Matrix Controller redundancy manager sends an *CCS Redundancy Report* message with every transition from one state to another.

The possible states reported are:

- 0x01 - Primary Active, Secondary Synchronizing
- 0x02 - Primary Synchronizing, Secondary Active
- 0x03 - Primary Active, Secondary Standby
- 0x04 - Primary Standby, Secondary Active

States 3 and 4 are stable, and the card pair should be in one of these states most of the time.

Constraint

The *CCS Redundancy Configure* message will fail if the ISDN card in the secondary slot already has D channels assigned to it. A status of 0xF2 “CCS Redundancy: Not the Primary Slot” returns. Only the ISDN card in the primary slot may have D channels assigned prior to the *CCS Redundancy Configure* message being sent. The host application must keep track of which ISDN cards currently have D channels assigned.

Scenario

CCS Redundancy is configured with the primary ISDN card in slot 6 and the secondary card in slot 7. The primary ISDN card fails, causing a switchover to the secondary card (and card redundancy to be deconfigured). All ISDN D channels are not assigned to the ISDN card in slot 6.

A new ISDN card eventually replaces the failed primary card in slot 6. If the host wishes to re-assign redundancy, it must issue the *CCS Redundancy Configure* message with the primary slot set to 7, (the card with the D channels currently assigned), and the secondary slot set to 6 (the newly replaced card). Note that this is the opposite way the cards were originally configured.

Disabling Redundancy

The CSP disables redundancy under the following conditions:

- The primary or secondary ISDN card is removed.
- A dead primary or secondary ISDN card is detected.
- The primary or secondary ISDN card is taken out of service.
- The ISDN Redundant I/O card fails, or the card is removed.
- The CCS I/O Series 3 card fails, or the card is removed.
- A Matrix Controller switchover occurs while the ISDN cards are synchronizing.
- A timeout occurs during synchronization. (The default time limit is 120 seconds.)

Important! When you remove an I/O card, the state of the secondary ISDN card changes from standby to single and all D channels on that card are de-assigned.

CAUTION

*Always press the **STOP** button on the ISDN Series 3 line card in a redundant card pair configuration before removing or inserting the corresponding I/O card.*

For example, in a redundant ISDN Series 3 card pair configuration, be aware that the redundant line card may reset when removing or inserting the primary I/O card. If this occurs, the system will lose calls.

You can disable the redundancy configuration from the host by sending the *CCS Redundancy Configure* message and setting the *Secondary Slot* field to 0xFF. This action changes the active and standby cards to an independent state and de-activates the D channels on both cards. If you do not configure redundancy when you send the message, the system still returns a positive ACK response.

Reconfiguring Redundancy

If you disable card redundancy by removing one of the cards, the remaining ISDN card processes calls as an independent card.

To activate redundancy after replacing the card, send the *CCS Redundancy Configure* message and designate the active card as primary and the replaced card as secondary. Dialogic recommends sending this message when the system is idle.

CAUTION

Configuring the re-installed card as the primary card resets all configuration and active call information on the active card.

If a switchover occurs when the cards are in state 3 (primary active, secondary standby) and the active card is reset, the following messages are sent to the host in the order indicated:

1. The *Alarm* message reports an alarm condition for the active card.
2. The *CCS Redundancy Report* message reports state 2 (primary synchronizing, secondary active).

3. The *CCS Redundancy Report* message reports state 4 (primary standby, secondary active).

Querying Redundancy

You can query the redundancy information by sending the *CCS Redundancy Query* message. The CSP returns the state of the redundant pair and identifies the primary, secondary, and active slots.

If a Matrix Controller switchover occurs while the two ISDN cards are synchronizing and they have not reached the active and standby states, the system resets both cards to the default configuration.

If the cards are already in active and standby states and the Matrix Controller becomes active, the *Card Status Report* and *CCS Redundancy Report* messages are sent to the host to indicate the state of the redundant cards.

ISDN Congestion Control

Overview

The ISDN PRI Congestion Control feature prevents the ISDN card from failing due to call traffic overload. The Congestion Control scheme will help reduce message processing undertaken by the ISDN card and other tasks within the system should incoming traffic on the D channels become excessive.

The ISDN PRI Congestion Control is enabled by default and Dialogic recommends that you do not disable it. The ISDN card measures the number of incoming messages on all supported D channels over set time periods. These congestion control parameters are configurable. If the threshold parameters are set too low, the ISDN card could reject calls that it would be capable of processing. If they are set too high, the CSP will be ineffective in preventing an excessive load situation.

Implementation

The level of congestion is monitored by three components in the ISDN protocol stack:

L3P Call Control component (0x05)

L3 Call Reference component (0x08)

L3 D Channel component (0x0A)

The congestion control parameters for the above components are configurable with the *ISDN Interface Configure* message (0x60). Congestion parameters configuration information may be retrieved using the *ISDN Query* message.

When the congestion thresholds are met, an *Alarm* message is sent to the host. Where Dialogic's default parameter values are used, congestion is cleared when the average message count is less than 300 messages for 5 seconds.

The Congestion Threshold data fields are common to both ISDN and DASS congestion control schemes. The data fields are formatted to ensure backwards compatibility with the existing DASS congestion control scheme. DASS congestion control used just a single congestion level, this has been changed to Level 2 burst and average thresholds.

The default values have also been changed to work with the ISDN scheme, if using DASS then Level 2 burst threshold should be set to 200 and Level 2 average threshold should be set to 1000.

Congestion Level

When the number of incoming messages reaches a certain pre-defined threshold, the D channels are identified as being in a congested state. A congested state results in any new incoming or outgoing call attempts being either rejected or ignored. The D channels will remain in the congested state until the incoming message rate falls below a pre-defined abatement threshold.

The two pre-defined levels of congestion exist with thresholds defined for each. Congestion Level 1 defines the threshold at which calls are rejected. Congestion Level 2, for the higher level, defines the threshold at which point calls are completely ignored.

The congested state is removed once the incoming message rate slows down to a pre-defined, lower threshold known as the abatement level, and stays below this level for a period of time. At this time, processing of the incoming and outgoing calls resumes.

Default Values

The table below shows the default settings for Congestion Control. These settings are applicable on a per ISDN card basis.

Description	ISDN PRI card Defaults	ISDN Series 3 card Defaults
Level 2 burst threshold	400 messages	700 messages
Level 2 average threshold	1900 messages	3500 messages
Abatement threshold	500 messages	900 messages
Burst time window	1 second	1 second
Number of samples in average	10 bursts	10 bursts
Abatement window	5 seconds	5 seconds
Level 1 burst threshold	350 messages	600 messages
Level 1 average threshold	1800 messages	3300 messages

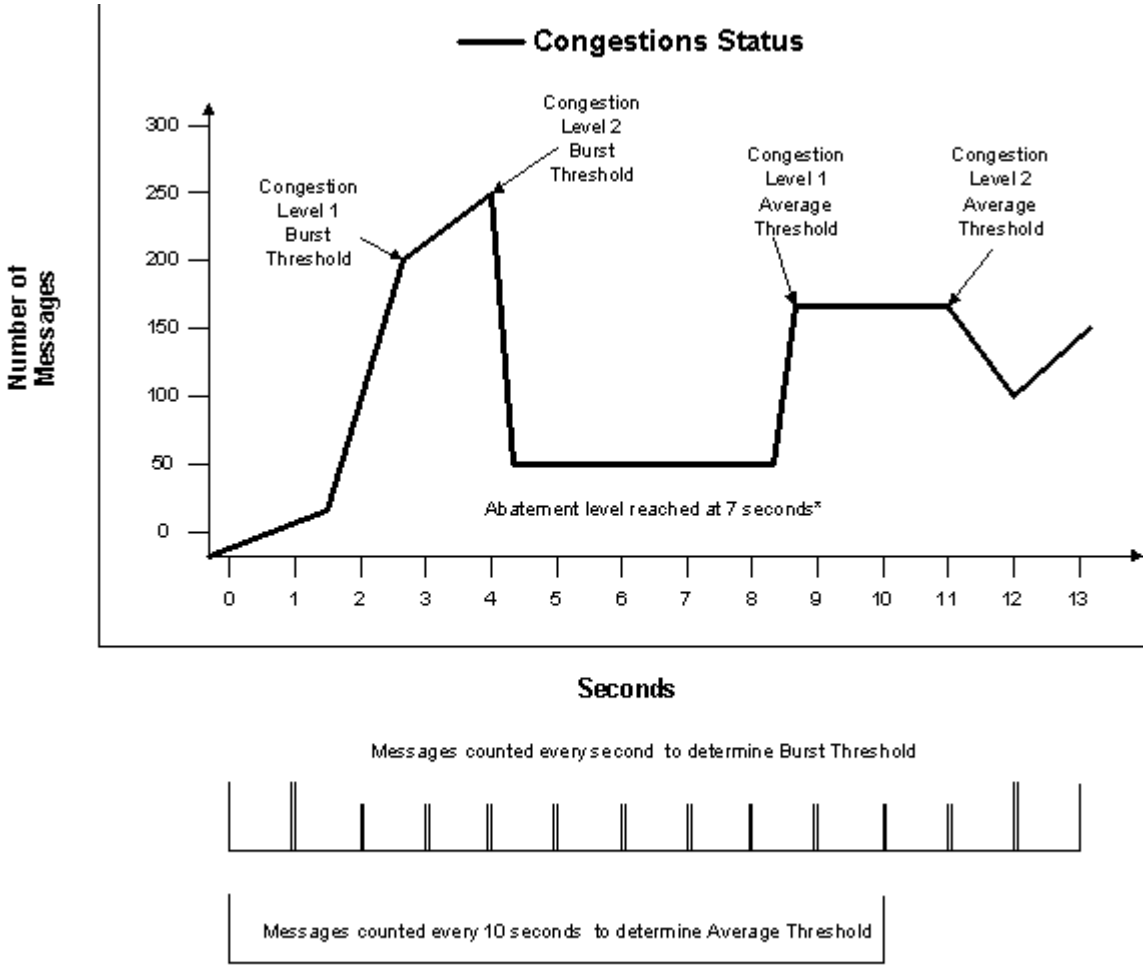
The ISDN congestion control scheme continuously counts the number of incoming messages on an ISDN card. Samples are taken in fixed units of time and are used to determine the congestion level of the card's associated D channels.

Individual samples are used to determine the 'burst' rate. A burst is a count of all incoming messages in a single sample. This enables short excessive peaks in traffic to be accounted for.

A combination of samples is used to determine a moving average. This moving average is a count of all incoming messages over a number of samples. This enables consistent excessive traffic to be accounted for.

Diagram The figure below shows congestion occurring at various levels using the default settings.

Figure 8-7 Example of Congestion



* Congestion is cleared when the average message count is less than 300 for five seconds.

Configuration

Congestion control on the ISDN card is configurable through the *ISDN Interface Configure* message. The Single Channel format AIB, (0x0D) is used in this message to specify any one of the individual D Channels but it will apply to all. The AIB should be used to address any individual D channels that have been previously configured using *D Channel Assign* (0xC4). A new entity, Congestion Threshold (0x0C) has been added to this message to provide congestion threshold values. These values can be changed.

No Congestion during Call Processing

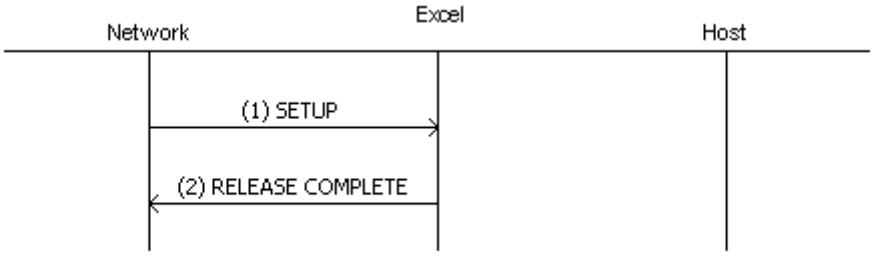
Calls are processed normally.

Congestion Level 1 during call processing

When Congestion Level 1 is reached, a Congestion Level 1 alarm is sent to the host in an *Alarm* message.

Incoming Calls

Incoming calls are rejected by the Layer 3 Call Reference component (0x08) and a RELEASE COMPLETE message is sent to the network with a Clearing Cause of 0x42 “switching equipment congestion” (0x42).

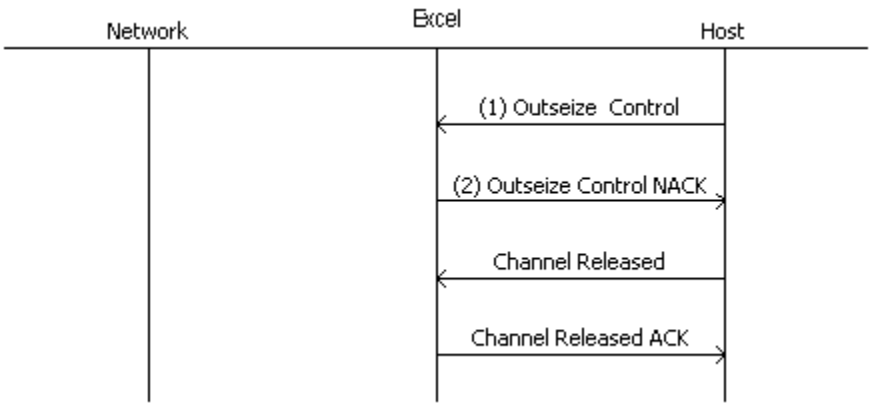


1. A SETUP message is sent from the network to the CSP after Congestion Level 1 is reached.
2. RELEASE COMPLETE with cause code of 0x42 “switching equipment congestion” is sent to the Network.

Important! Calls initiated using the REGISTER message are rejected in the same way.

Outgoing Calls

Outgoing calls are rejected by L3P Call Control component (0x05). On receipt of the *Outseize Control* message, a NACK is sent to the host with a status of “ISDN congestion” (0x53).



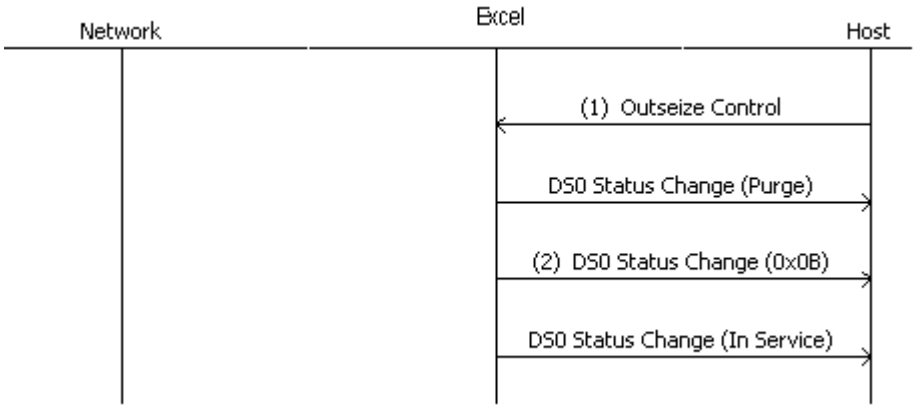
1. When Congestion Level 1 is reached, the host sends an *Outseize Control* message to the CSP.
2. Outseize Control NACK has a status of 0x53 which indicates Congestion Level 1 has been reached.

Congestion Level 2 during call processing

Incoming Calls

Incoming calls are discarded within the Layer 3 D channel component (0x0A). All messages initiating new calls will not be processed.

Outgoing Calls



1. When Congestion Level 2 is reached, an *Outseize Control* message is rejected. The host then is sent a *DS0 Status Change* of Purge.
2. The *DS0 Status Change* message is sent to the host with a Purge Reason of 0x0B, FEM Outseize Acknowledgment Timeout.

ISDN Bearer Connection Independent Supplementary Services

Overview

Provision for ISDN Bearer Connection Independent Supplementary Services is implemented using the FACILITY message through the ISDN GCR component (0x09) with optional facility information element (IE) contents. All that is required for these services is the ability to transmit and receive FACILITY messages with a dummy call reference IE.

Existing support for FACILITY messages and both *normal* and *global* Call Reference IEs are not affected. ISDN FACILITY messages are sent from the host using the *PPL Event Request* (0x44). The host is notified of the receipt of the incoming ISDN FACILITY message using *PPL Event Indication* (0x43).

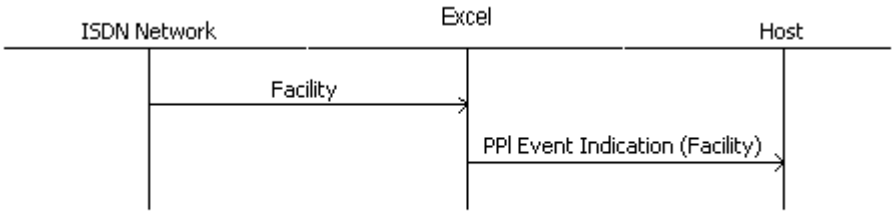
Implementing incoming calls

Implementation of ISDN Bearer Connection Independent feature for incoming calls requires the following configuration:

- Enable the sending of the FACILITY INDICATION message to the host.
- Set PPL Config Byte 3 of the Global Call Reference component (0x09) to 0x01 (Enable) using *PPL Configure* (0xD7)

Example incoming call flow

The following indicates the call flow for an incoming FACILITY message with dummy call reference:



1. The network sends a FACILITY message with CR = null to the CSP.
2. The CSP sends a *PPL Event Indication* (FACILITY) to the host.

Important! The Call Reference IE ICB is not sent to the host in the *PPL Event Indication* message.

Implementing outgoing calls

The Call Reference Information Element is used in the ISDN Formatted IEs ICB in the *PPL Event Request* message to send a Call Reference value to the CSP. The format of the ISDN Formatted IEs ICB and the Call Reference IE are shown below:

ISDN Formatted IEs
(Subtype 0x10)

ICB Data Length	Variable
ICB Data[0]	Number of IEs (0–30)
ICB Data[1]	IE 1 Type
ICB Data[2]	IE 1 Length (0–130)
ICB Data[3]	IE 1 Data[0]
:	:
:	
ICB Data[:]	IE n Type
ICB Data[:]	IE n Length (0–130)
ICB Data[:]	IE n Data[0]

Call Reference IE Format
Data[0] IE Type 0x01
Data[1] Length
Data[2+] Data

To request a dummy call reference, enter the following:

Data[0] IE Type 0x01 Call Reference
Data[1] Length - 0x00
Data[2] There is no data

To send a 1-octet Call Reference value, enter the following:

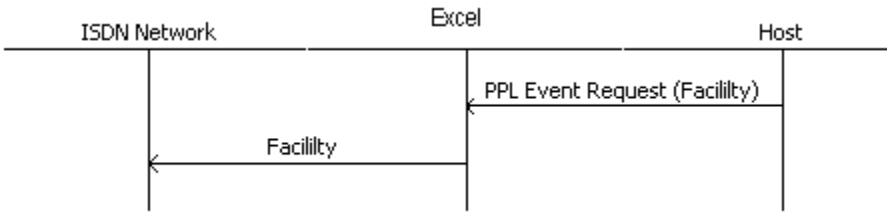
Data[0] IE Type 0x01 Call Reference
Data[1] Length - 0x01
Data[2] Call Reference value

To send a 2-octet Call Reference value, enter the following:

Data[0] IE Type 0x01 Call Reference
Data[1] Length - 0x02
Data[2] Call Reference value octet 1
Data[3] Call Reference value octet 2

Example outgoing call flow

The following indicates the call flow for an outgoing FACILITY message with dummy call reference.



1. The host sends a PPL Event Request (FACILITY message) to the CSP with a Formatted IEs ICB [null CR IE].
2. The CSP sends to the network a FACILITY message with CR = null.

Link Access Protocol D-Channel (LAPD)

Through a low-latency, high-speed interface, Dialogic has exposed the Link Access Protocol D-Channel (LAPD) directly to the host application, bypassing Layer 3 and Layer 4. For direct communication with the ISDN Series 3 card, the host connects to the CCS I/O card with 10-BaseT Ethernet. A new CCS I/O Series 3 card or "packet engine" has been introduced to terminate Layer 2 (Q.921) only—the card does not have Layer 3 or Call Control capability. Packets are not sent to the host using the CSP Matrix Series 3 Card. Instead, they are sent from the CCS I/O Series 3 (using the local area network) to the LAPD host. Configuration and alarms are sent only to the matrix host.

> **Important!** LAPD requires a product license. For details, refer to *Downloading License Keys to the CSP* in the *Licensing Overview* chapter in the *Developer's Guide: Overview* and the *Product License Download* message (0x0079) in the *API Reference.*

To enable direct communication with the host through the CCS I/O card, changes were made to the Layer 2 interface, Layer 2 management, configuration, and internal communications. All pre-existing Layer 2 functionality is still supported. ISDN Layer 2 resides on the ISDN Series 3 card.

Support for Transmission Control Protocol (TCP) over IP has been added to CCS cards, with a two-byte sequence number. TCP messaging is used to optimize response time to and from the local host. A message must first be received from the host so that the port number can be stored for use in subsequent indications. Dialogic has also simplified the Subrate point-to-multi-point connect and disconnect messages.

Dialogic's LAPD design allows for a maximum of 65,536 sequence numbers on the CCS I/O Series 3 card. The sequence numbers indirectly have an impact on the retry mechanism for switch-based (service card) messages. The retry mechanism allows the service card to resend any unacknowledged messages to the host on a regular interval for a defined number of retry times. With the use of the direct TCP/IP link between the service card and the host, a high volume of messages can be processed in a given time period, which in turn will often result in a large number of outstanding unacknowledged messages.

The host designates which E1 or T1 spans are assigned D channels (up to 128), and which D channels have what Service Access Point Identification (SAPI) and Terminal Endpoint Identification (TEI) values. The host can configure both the network side and the user-side Global System Mobile (GSM 8.56) Terminal Endpoint Identifiers (TEIs). Each D channel supports eight TEIs, and the Matrix system management logic has been modified so that 128 D channels can be configured on a single CSP node.

The LAPD entities serve as the network side, and communicates with the user side's wireless base stations and transceivers. The signaling information is transmitted as LAPD packets over DS0 channels of E1 links. The LAPD can also serve as the user side.

LAPD Redundancy

For information on LAPD redundancy See *ISDN Redundancy (8-69)*.

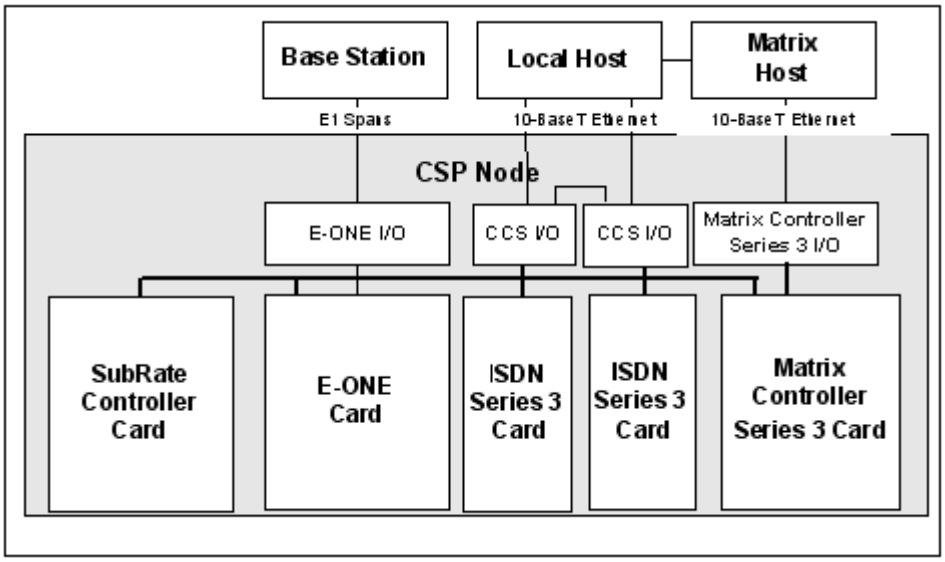
LAPD Limitations

- Management of card-level congestion for overload conditions is not supported
- Removing the CCS I/O card causes the ISDN Series 3 standby card to reset. Redundancy is dropped and the active card becomes single.
- Removing the CCS I/O of the ISDN Series 3 Active card causes a switchover to the ISDN Series 3 secondary card.
- You must remove both cards and re-insert if you are performing maintenance on either card.

LAPD Architecture Overview

Overview The figure below is an overview of the CSP architecture. The local host and matrix host are depicted as separate entities, but they can also exist as a single entity.

Figure 8-8 An overview of the CSP, with hosts connecting the ISDN and CSP Matrix Series 3 Cards

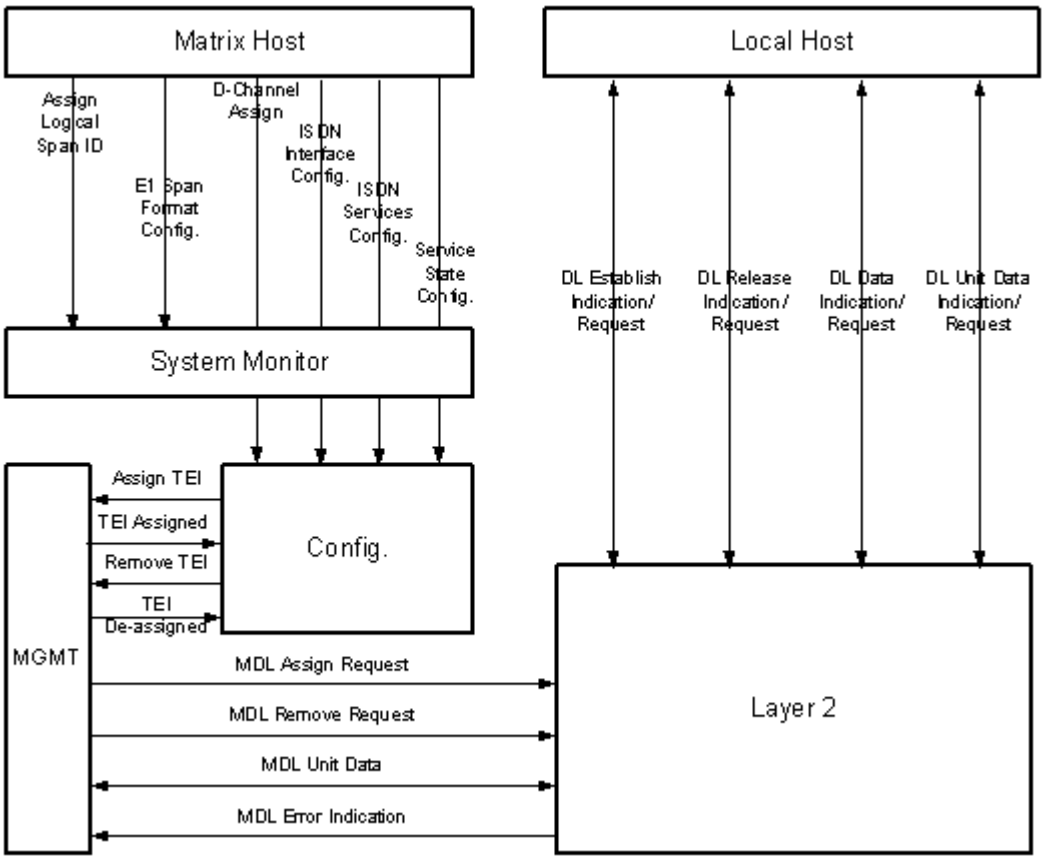


LAPD Functional Message Breakdown

Overview The figure below shows the messages and tasks for enabling Layer 2, the TEI management messages, and the interface to the host. All Layer 2 messages to and from the local host are embedded in these two messages:

- PPL Event Indication
- *PPL Event Reques*t.

Figure 8-9 Input/Output Messaging



LAPD Configuration Sequence

Overview The following is the configuration sequence to bring LAPD channels into service to communicate with the host.

1 De-assign Logical Span IDs
API message: *Assign Logical Span IDs*

2 Assign Logical Span IDs to the physical E1 spans
API message: *Assign Logical Span IDs*

3 Configure E1 Format to Clear Channel
API message: *E1 Span Configure*

4 Assign D channels to the ISDN Card slot
API message: *D Channel Assign*

5 Configure the Connection Endpoint type to be Euro-ISDN
API message: *ISDN Interface Configure*

6 Configure whether LAPD Entity is User or Network side
API message: *ISDN Interface Configure*

7 Configure the D channel Speed (default is 64 kbps)
API message: *ISDN Interface Configure*

8 Configure the LAPD options to enable host-to-Layer 2 messaging
API message: *ISDN Terminal Configure*

9 Configure the terminal types and TEI values
API message: *ISDN Terminal Configure*

10 Configure LAPD timers and counters (if it is necessary to change them)
API message: *ISDN Terminal Configure*

11 Bring spans into service
API message: *Service State Configure (spans)*

12 Send PPL Event Request to L2 to redirect Output to Local Host
API message: *PPL Event Request (Layer 2)*

13 Bring the D channels into service
API message: *Service State Configure*

14 Await Layer 2 up DL Establish. This and subsequent Establish Indications are reported to the Local Host.
API message: *PPL Event Indication (Layer 2).*

LAPD Inter-Module Configuration and Startup

Overview

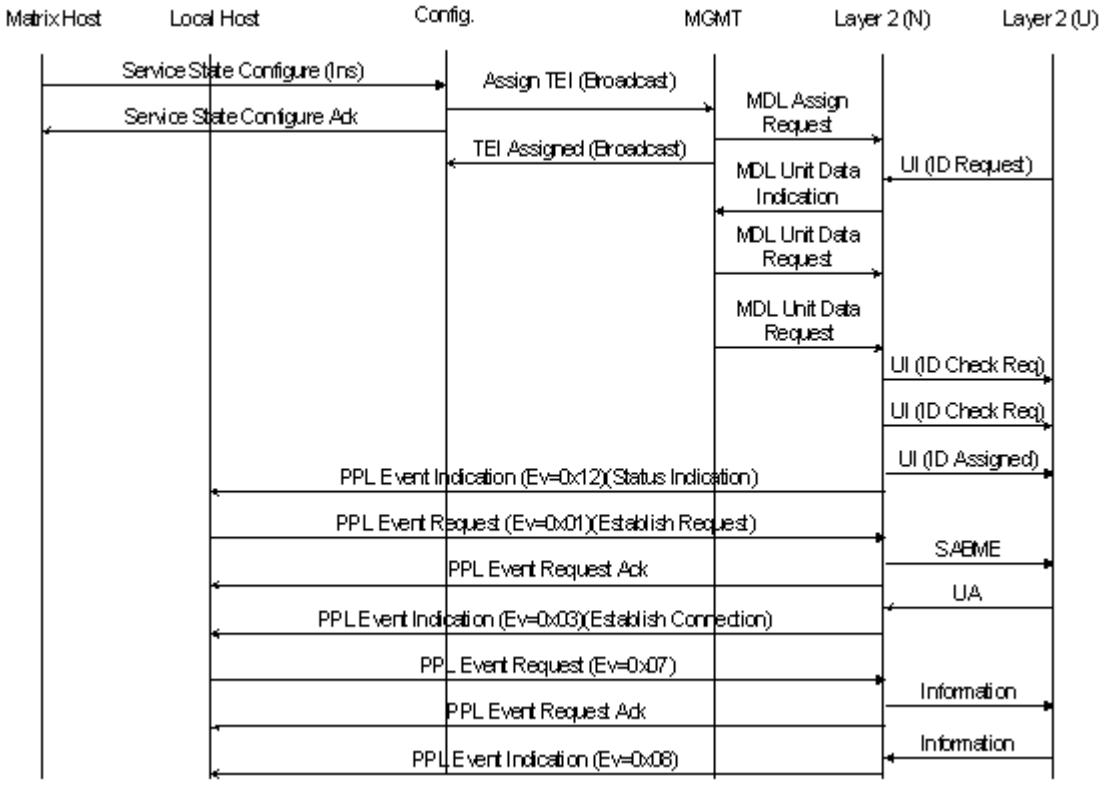
The figure on the next page represents host messaging when a network-side LAPD connection is brought up (already correctly configured) with GSM 8.56 TEIs. Upon receipt of the *Service State Configure* message, the Broadcast TEI is assigned. Management then waits for the ID request to be received from the user side.

After the software management (MGMT) system receives the ID request, the Transceiver (TRX) has the Action Indicator (AI) set to the TEI that it wants to establish (must be in the range 0-63). The network side sends an ID Check Request and waits for Timer 201 (T201) to expire. If T201 expires before it receives an ID Check Response, the network-side resends the Check Request and restarts T201. After this expiry, MGMT sends the ID Assigned, and updates its tables. When the TRX sends the ID Request, it starts T202, which is longer than T201. T202 expiration implies a failure, and the user terminal resends the ID Request.

After T202 attempts to get a TEI, the Layer 2 manager sends the host a *PPL Event Indication* message indicating an assignment failure.

Once the TEI assignment is successful, the host receives the *PPL Event Indication* message with status indication, Event=0x12, indicating that the DLCI state has changed to the TEI assigned state. The host must then issue the Establish Request PPL Event Request (Event=1) to bring up the data link. Once it is established, the *PPL Event Indication* message (Event=3) will be sent to the host indicating Establish Confirm. Upon reception of the *Service State Configure* message, the Broadcast TEI is assigned. Management then waits for the ID request to be received from the user side.

Network-side LAPD connection brought up with GSM 8.56 TEIs

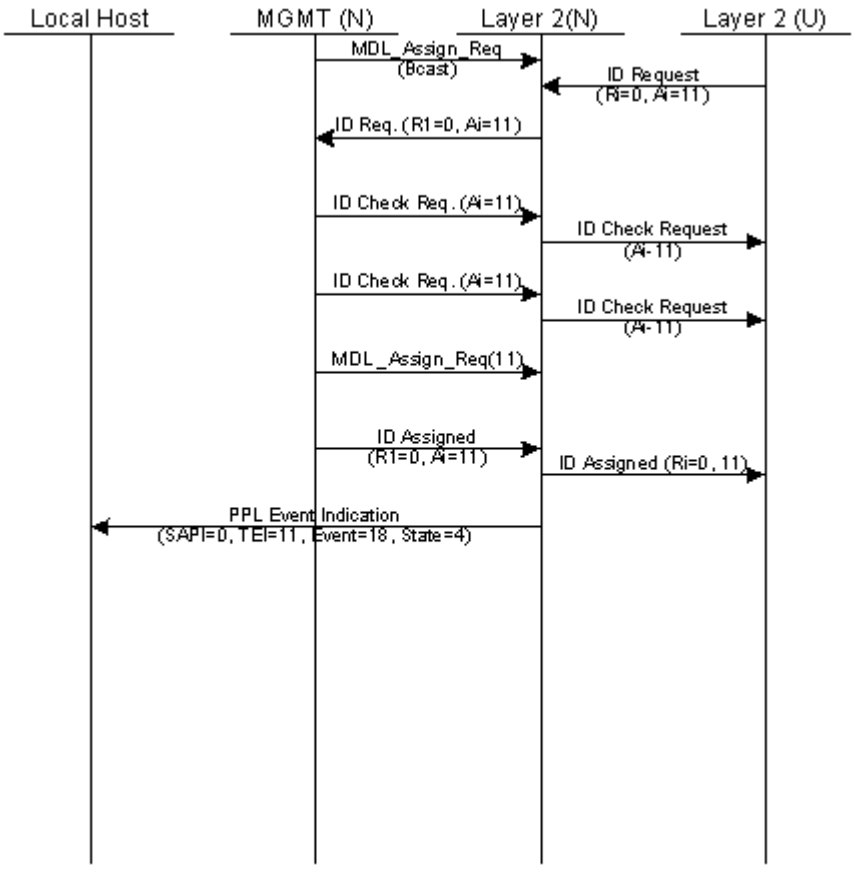


The figure above shows the user-side requesting an ID with the non-standard Ri=0 and the AI=11 (11 was chosen at random for the example). MGMT receives this message and checks for TEIs configured for GSM 8.56. If the AI requested was configured by the host, then MGMT starts the check procedures (non-standard). When performing the checking procedures, it starts T201. In this case there are no other terminals with that TEI assigned, so there is no response. Upon T201 expiry, another *ID Check Request* message is sent, and if no response occurs, on that expiry the *ID Assigned* message is sent to the user. The result is that both Layer 2s for SAPI=0, and TEI=11 are in State 4 which is TEI-ASSIGNED. There is also a *PPL Event Indication* with the Status ICB with status value of 4 sent to the host.

When the user side sends the ID Request, T202 is started. In the above example, all of the messaging occurs prior to T202 expiring. When the ID Assigned message is received, T202 is stopped. Note that there are

times when the Layer 2 Data Link Connection Identifier (DLCI) state machine can return to State 4 (TEI Assigned) and the host must issue another DL Establish Request to bring up the DLCI again.

Normal Startup Procedure



LAPD and Host Communication

Establish a connection

To establish a connection link from the host to the switch, refer to the *Hardware Installation and Maintenance Guide*. Once you have connected to a communications link, obtain an Internet Protocol (IP) address. For more information see the *API Developer's Guide: Overview.*

IP Address and Subnet Mask

The IP address is a 32-bit address used in IP routing. IP addresses and subnet masks can be assigned either through Reverse Address Resolution Protocol (RARP), Bootstrap Protocol (BOOTP), or through the *IP Configure* message.

When the ISDN card starts up and finds an invalid IP address or subnet mask stored in the Electrically Erasable Programmable Read-Only Memory (EEPROM), it will issue five RARP requests followed by five BOOTP requests. If the card does not contact either a RARP or BOOTP server, then the card will start up before Transmission Control Protocol/Internet Protocol (TCP/IP) has started. If the card is successful in contacting a RARP or BOOTP server, then the server's values will be used for an IP address and subnet mask and TCP/IP is started. To store an invalid IP address and subnet mask in the EEPROM, use an *IP Configure* message with the IP address set to 255.255.255.255 and any value for the subnet mask.

When the ISDN Series 3 card starts up and it finds a valid IP address and subnet mask are stored in the EEPROM, both RARP and BOOTP attempts are skipped and TCP/IP is started with the IP address and subnet mask found in the EEPROM. The *IP Configure* message is the only method to set the IP address and subnet mask values in the EEPROM.

If the host sends a *IP Configure* message and the IP address and subnet mask in the message are different than the values stored in the EEPROM, the EEPROM will be overwritten with the new values. The card will need to be rebooted for the new IP address and subnet mask to take effect.

If the host assigns the same IP address and subnet mask, the message is acknowledged and the card does not have to be reset. This allows the host to send the *IP Configure* message whenever it considers it necessary.

The subnet mask defaults to the values shown in the following table for BOOTP and RARP. Make sure that cards are in the same subnet.

Class	IP Address	Subnet Mask
A	1.0.0.0 - 127.0.0.0	0xFF.00.00.00
B	128.0.0.0 - 191.0.0.0	0xFF.FF.00.00
C	192.0.0.0 - 254.0.0.0	0xFF.FF.FF.00.

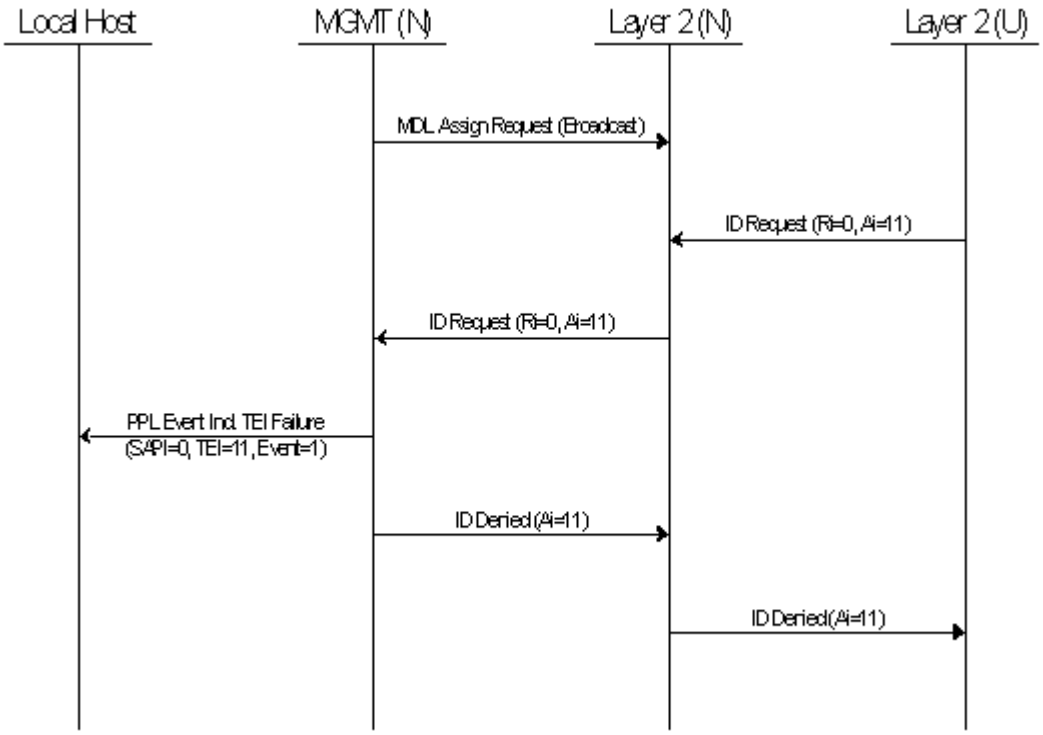
Incorrectly assigning a class to the switch may result in superfluous messaging that can adversely affect system performance. If a switch receives a broadcast message from another switch using a different class, it interprets the message as being addressed specifically to the switch using an incorrect IP address.

The switch responds by sending an Address Resolution Protocol (ARP) request to determine which node originated the message. The switch then sends an Internet Control Management Protocol (ICMP) message to indicate that the node ARP cache has an incorrect IP address stored for the switch. The other node clears the switch IP address from its ARP cache. The next time that node attempts to send a message to the switch, it sends out an ARP request to get the switch IP address.

LAPD Example Call Flows

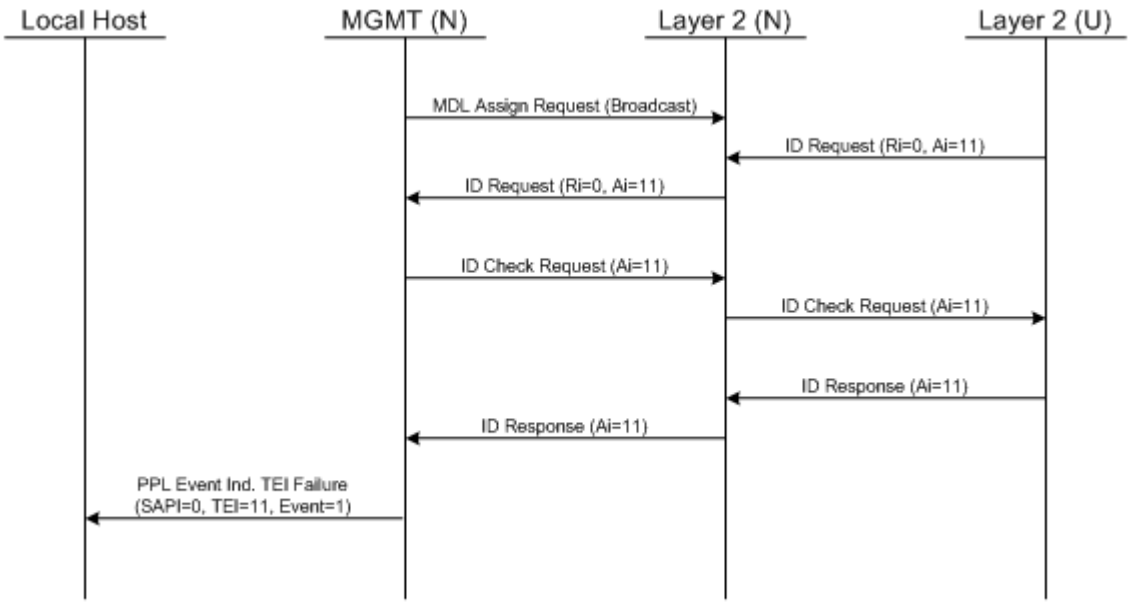
TEI Denied Procedure

The procedure below shows an ID Request from the user side, requesting TEI 11. Management checks the Configuration database but cannot find TEI 11 listed, so it issues an ID Denied. Management sends a *PPL Event Indication* message to the local host to indicate that this TEI failed assignment procedures.



TEI Already Assigned Procedure

The procedure below shows a check procedure being initiated when a TEI is already assigned. The *ID Check Response* message returns the same TEI, and the TEI failure is propagated to the host.

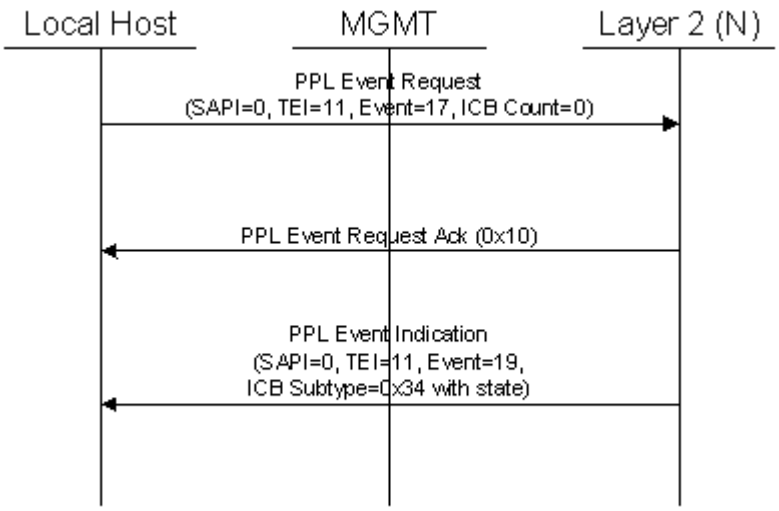


Host State Query

The figure below illustrates a query for the state of a specific TEI/SAPI combination on a D channel. The *PPL Event Request* message contains the following:

- LAPD Status ICB
- addresses a slot
- D channel number
- SAPI
- TEI

The result appears in a *PPL Event Indication* message containing an LAPD Status ICB.


Local Host
MGMT
Layer 2 (N)
PPL Event Request
(SAPI=0, TEI=11, Event=17, ICB Count=0)
PPL Event Request Ack (0x10)
PPL Event Indication
(SAPI=0, TEI=11, Event=19,
ICB Subtype=0x34 with state)

Subrate Switching

Overview

The Subrate Switch Controller (SRC) card allows one-way and two-way non-blocking connections and disconnections between any two subrate channels in the CSP.

Configuration

The SRC card requires no configuration. Once the card is inserted into the chassis, it is ready for call processing.

Redundancy

SRC redundancy is achieved using two SRC cards—one active and one standby. When a second SRC card is inserted into the chassis, it is automatically configured as a standby card. No configuration by the host is required. If the active card fails, the standby card automatically establishes control of all subrate connections, with no interaction from the host. The switch supports two SRC cards.

Connection Management

Subrate connections and disconnections are managed using the *Subrate Connection Management* message. Subrate channels are defined as any number of sequentially ordered bits in an 8-bit DS0.

Multiple subrate channels can be established within a single DS0, as shown in the table below. The host manages connections to ensure that subrate channels within a single DS0 do not overlap (That the same bit is not being used in two different subrate channels).

Table 8-1 Subrate Channels per DS0

Channel Size (bits)	Number of Channels
1	8
2	4
3	2
4	2
5	1

Connection Scenarios

Both subrate channels in a subrate connection must be the same size. The number of connections supported per subrate channel size is shown in the table below.

The SRC card can simultaneously manage connections of differing rates. For example, a two-bit-to-two-bit connection could be made between any two subrate channels, while a one-bit-to-one-bit connection is made on two other subrate channels.

The number of simultaneous connections supported depends upon the combination of subrate channel sizes used. For example, a single switch can simultaneously connect 1,024 8-bit channels (512 two-way connections) and 4,096 two-bit channels (2,048 two-way connections).

Connection Types

The SRC card supports both two-way and one-way subrate connections. The connection type is determined by the Action field in the *Subrate Connection Management* message. Multiple one-way connections with the same source subrate channel can be used to establish a broadcast, as shown in the Examples section of the message, Subrate Connection Management 0x0D 3-74, in this document.

Disconnection

The host is responsible for tearing down subrate connections. The switch cannot tear down a subrate connection for any reason, including the loss of a span, except in response to a *Subrate Connection Management* message from the host.

Table 8-2 Connections supported per subrate channel size

Subrate Channel Size (bits)	Rate	Number of Connections
8	64 Kbps	1,024
7	56 Kbps	1,024
6	48 Kbps	1,024
5	40 Kbps	1,024
4	32 Kbps	2,048
3	24 Kbps	2,048
2	16 Kbps	4,096
1	8 Kbps	8,192

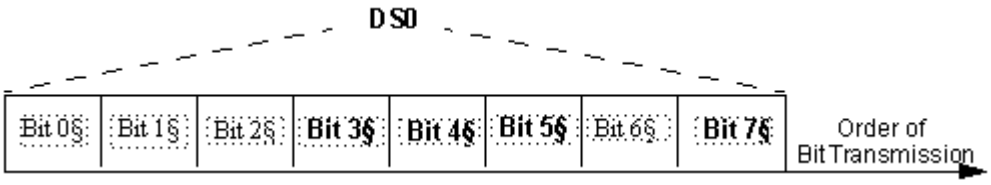
Subrate Connection Management Message

The *Subrate Connection Management* message is used to establish subrate connections and to tear them down. The *LSB of Subrate Channel* field in this message indicates the first low-order bit of the subrate channel in the DS0. Bits are transmitted Most Significant Bit (Bit 7) to Least Significant Bit (Bit 0) as shown in the figure below.

The *Number Of Bits In Subrate Channel* field in this message indicates the number of bits (0-7) in the subrate channel. To ensure that the subrate channel does not overlap another DS0, the number of bits in the subrate channel plus the Bit Offset must not exceed eight.

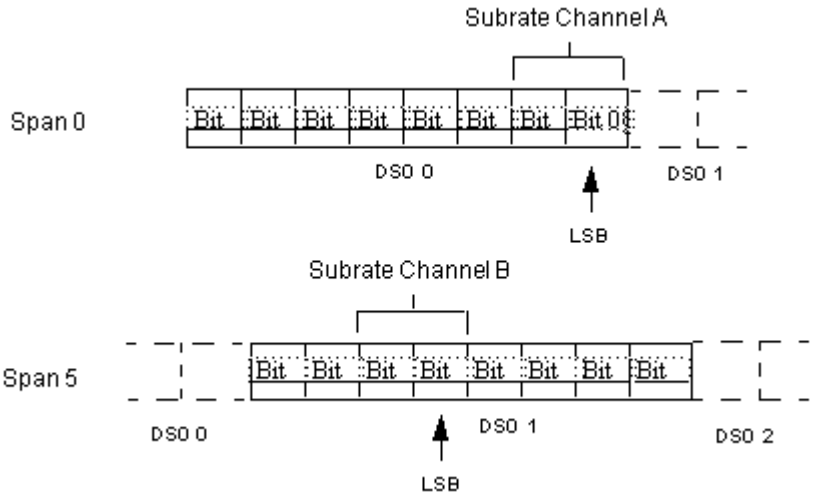
The *Action* field in this message indicates a one-way connection (0x01), a two-way connection (0x02), or a disconnection (0x03).

Figure 8-10 Order of Bit Transmission



Examples This section includes examples of one-way and two-way connections and disconnections using the subrate channels shown in the figure below.

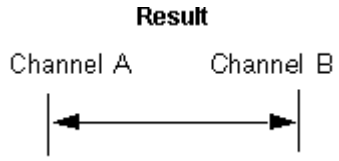
Figure 8-11 Subrate Channel A and B



Two-Way Connection

The figure below shows the establishment of a two-way connection between the two subrate channels.

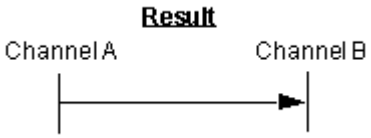
Figure 8-12 Two-Way Connection - Result



One-Way Connection

The figure below shows how a one-way connection is established between the two subrate channels. In this example, subrate Channel A is the transmission source for subrate Channel B.

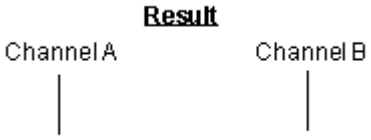
Figure 8-13 One-way subrate connection - Result



Two-Way Disconnection

The figure below shows the use of the API format of the *Subrate Connection Management* message to tear down an established connection. The message is identical to that used to establish the connection except that the *Action* field (Byte 24) is set to *Disconnect* (0x03) as shown in the next table.

Figure 8-14 Two-Way Disconnection - Result

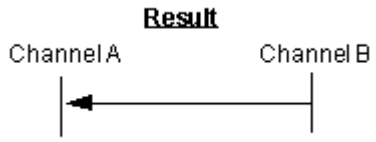


One-Way Disconnection

For a one-way disconnection, the same subrate channel is entered as both Subrate Channel A and Subrate Channel B in the message.

If a one-way connection is established, it causes termination of the connection at both ends.

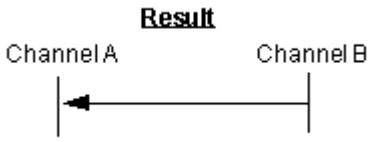
Figure 8-15 One-Way Disconnection - Result



If a two-way connection is established, it results in a one-way connection with the subrate channel indicated in the message as the source channel.

The figure below shows teardown of a two-way connection through the use of the *Subrate Connection Management* API message. This action results in a one-way connection with Subrate Channel B as the source.

Figure 8-16 Two-Way Connection - Result



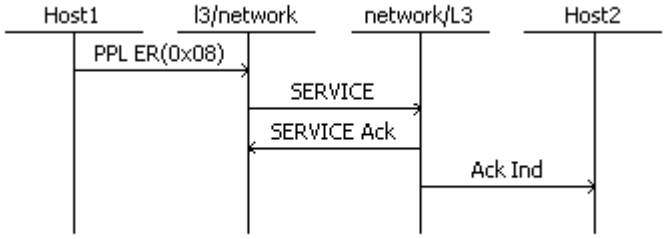
Support for ISDN Busy Out Channel with PPL Event Request

Description

This feature allows the host to busy out the channel though a *PPL Event Request* message for the 4ESS variant of ISDN in the BSS component.

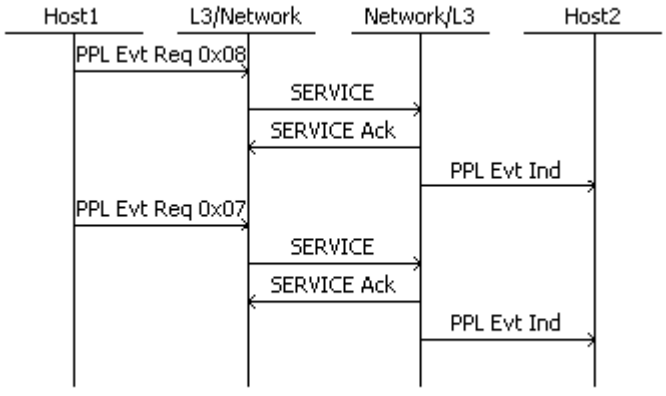
Busy Out B Channel

When the host sends the Event Request, the CSP does not disconnect the call. Instead it busies out the B channel. The host will receive a positive ACK. The host sends another Event Request to bring that channel back into service for subsequent calls.



Configuration Byte to Send Event Indication

The configuration byte 0x03 in the BSS component allows this component to send the Event Indication to the host reporting the B channel state.



Event Indication Sent

When the B channel is busied out, the BSS Component reports that to the host with PPL Event Indication 0x02.

Refer to *L3 B Channel Control (0x000B) (9-17)* for the details of the PPL information involved with this feature.

ISDN Over Ethernet

Overview

Prior to this release, the communication between the ISDN Series 3 card and the CSP Matrix Series 3 Card used the mid-plane HDLC bus. Communication over the HDLC bus slows the maximum number of messages that can be transmitted to be less than 100 per second, not allowing the user to achieve high call rates that require ISDN services.

Using Ethernet communication for messaging between a ISDN Series 3 card and CSP Matrix Series 3 Card allows increased performance by providing more message transactions per second.

This feature supports a new RCOMM state machine on the ISDN Series 3 card that monitors the status of the link between this card and CSP Matrix Series 3 Card. Communication over the HDLC bus will continue to be supported. Licensing of this feature is not required.

Description

When using the Ethernet communication mode, all the I/O ports are connected to the same physical Ethernet switch/hub. Redundancy at the port level is provided by the RCOMM state machine that monitors the status of the link between the ISDN Series 3 and Matrix Series 3 ports. The message packets are sent out of the port which indicates a link-up state. Additional logic provides a fall-back to the HDLC option in the event of a failure of all the Ethernet ports.

ISDN Series 3 Card and CSP Matrix Series 3 Card Ethernet Interconnections

API Alarm Monitoring

The *Alarm* (0x00B9) message supports the new 0x02 Card Alarm, 0x52 ISDN Ethernet Unavailable:

- If at any point the message communication mode is changed from Ethernet to the HDLC bus, an alarm is sent to the host.
- An Alarm Cleared (0x00C1) message is sent if the communication mode is changed back to the Ethernet mode.

9 ISDN PPL Information

Purpose This section contains Integrated Services Digital Network (ISDN) component-specific information for Configuration Bytes, PPL Timers, and PPL Events.

For a list of the PPL Components and addressing information, see PPL Component IDs and PPL Component Addressing in the *API Reference*.

Guidelines for Changing ISDN PPL Information

Overview

PPL timer and configuration bytes are stored on a ***per D channel*** basis for ISDN configuration. If a timer or configuration byte is changed for a D channel, all B channels controlled by the D channel will use the new timer or configuration byte value.

If you change the type of ISDN connection using the *ISDN Interface Configure* message, the system resets all B channel configuration parameters back to their default values. If you have to modify the default ISDN PPL configuration bytes or PPL timers, you must use the *PPL Configure* message ***after*** you change the Connection Type.

PPL Events

The following API messages are used to send and receive call processing PPL events:

- *PPL Event Reques*t

  This message is initiated by the host, sends an external

  event directly to the specified PPL component.

  The message can include the PPL component, PPL event, ICB

  count, ICB type, ICB subtype, and data.

ICB subtype values can include:

- PPL general purpose register data
- ISDN formatted IE
- ISDN raw IE

• *PPL Event Indication*

  This message enables the PPL component of the CSP to report an external event directly to the host application.The message can include the PPL component, PPL event, ICB count, ICB type, ICB subtype, and data.

ICB subtype values can include:

- PPL general purpose register data
- ISDN formatted IE
- ISDN raw IE
- PPL argument 2 data

L3P Call Control (0x0005)

Purpose L3P Call Control uses configuration byte and L3P/L5 event information.

The asterisks (*) denote default values.

L3P Call Control Configuration Bytes

Byte	Description	Values
0x01	Request for Service Format	* 0x01 = BCD Encoded 0x03 = Formatted ICB 0x04 = Exact Frame
0x02	Send ALERTING Request to L3P	* 0x00 = Send 0x01 = Do Not Send
0x03	Send DISCONNECT INDICATION to L5	* 0x00 = Disable 0x01 = Enable
0x04	Send ALERTING INDICATION to host	* 0x00 = Disable 0x01 = Enable
0x05	Send PROGRESS to host	* 0x00 = Disable 0x01 = Enable
0x06	Send CONNECT to host	* 0x00 = Disable 0x01 = Enable
0x07	Send RELEASE (COMPLETE) to host	* 0x00 = Disable 0x01 = Enable
0x0C	Send Call Proceeding event to host upon receipt of CALL PROCEEDING indication from network	* 0x00 = Disable 0x01 = Enable
0x0E	Send Outseize ACK event to host upon receipt of CALL PROCEEDING indication from network	* 0x00 = Disable 0x01 = Enable
For Euro-ISDN only:		
0x09	Overlap Receive mode	* 0x00 = Disable (False) 0x01 = Enable (True)
0x0A	Overlap Receive Supported Digit Length	10 digits
For Euro-ISDN and Austel only:		
0x0B	Send Setup ACK and More Info event to host upon receipt of SETUP ACK indication from network	* 0x00 = Disable 0x01 = Enable

Byte	Description	Values
0x0D	Send Outseize ACK event to host upon receipt of SETUP ACK indication from network.	* 0x00 = Disable 0x01 = Enable

3P/L5 Events

PPL Event Request (L5 to L3P)	PPL Event Indication (L3P to L5)
0x0001 ISDN Alerting 0x0002 ISDN Progress 0x0003 ISDN Connect 0x0004 ISDN Disconnect 0x0006 ISDN User Information 0x0007 ISDN Facility 0x000A † ISDN Register 0x000C ISDN Notify 0x000D ISDN Information 0x0020 ISDN Vari-A-Bill 0x0021 ISDN ANI-on-Demand	0x0001 * ISDN Alerting 0x0002 * ISDN Progress 0x0003 * ISDN Connect 0x0004 * ISDN Disconnect 0x0005 * ISDN Release (Complete) 0x0006 ISDN User Information 0x0007 ISDN Facility 0x0008 ISDN Facility ACK 0x0009 ISDN Facility Reject 0x000A † ISDN Register 0x000C ISDN Notify 0x000D ISDN Information 0x0013 More Information (L3P to L3) 0x0014 Call Proceeding Indication (L3P to L3)
* Default values † For the PPL Event Request register, the logical span ID and channel must indicate the D channel.	

L3P Call Control PPL Timers (0x0005)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Outseize Response Wait	60 seconds (6000 10ms units)
0x02	Register Request Wait	8 seconds

Timer ID	Description	Value (default)
For Euro-ISDN only		
0x02	INFO message wait for Overlap Receive	20 seconds
0x03	INFO message wait for Overlap Send	20 seconds
0x04	Register Request Wait	8 seconds

L3P Call Control PPL Events

Event Number	Event Name
0x50	L4 Call Request
0x51	L4 Alerting Request
0x52	L4 Connect Request
0x53	L4 Internal Connect Request
0x54	L4 Clear Request
0x55	L4 Inseize Control
0x56	L4 Outseize Control
0x57	L4 Purge
0x64	L3 SETUP Indication
0x65	L3 INFORMATION Indication
0x66	L3 CALL PROCEEDING Indication
0x67	L3 ALERTING Indication
0x68	L3 PROGRESS Indication
0x69	L3 CONNECT Indication
0x6A	L3 USER INFORMATION Indication
0x6B	L3 DISCONNECT Indication
0x6C	L3 Clear Indication
0x6D	L3 Error Indication
0x6E	L3 FACILITY Indication

Event Number	Event Name
0x6F	L3 FACILITY ACK Indication
0x70	L3 FACILITY REJ Indication
0x71	L3 Facility Req Ack Indication
0x72	L3 NOTIFY Indication
0x79	L3 REGISTER Indication
0x7A	L3 MORE INFO Indication
0x7B	L3 RELEASE Indication
0x96	L3_GEN_EVENT_IND_0
0x97	L3_GEN_EVENT_IND_1
0x98	L3_GEN_EVENT_IND_2
0x99	L3_GEN_EVENT_IND_3
0x9A	L3_GEN_EVENT_IND_4
0x9B	L3_GEN_EVENT_IND_5
0x9C	L3_GEN_EVENT_IND_6
0x9D	L3_GEN_EVENT_IND_7
0x9E	L3_GEN_EVENT_IND_8
0x9F	L3_GEN_EVENT_IND_9

L3P D Channel Control (0x0006)

Purpose This section contains the PPL Events for L3P D Channel Control.

L3P D Channel Control PPL Events

Event ID	Event Name
0x01	Force Initiation of DLC Establishment

L3P B Channel Control (0x0007)

Purpose This section describes the configuration bytes and atomic functions for L3P B Channel Control.

L3P B Channel Control Configuration Bytes

Byte	Description	Value
0x01	Number of B Enable Retries	0x02

PPL Events

Event ID	Event
0x0A	CFG In Service
0x0B	CFG Out of Service
0x0C	L1 Alive
0x0D	L1 Dead
0x0E	D Alive
0x0F	D Dead
0x10	B In Service
0x11	B Out of Service
0x12	B Maintenance
0x13	CR Purge

L3 Call Reference (0x0008)

Purpose

L3 Call Reference uses configuration byte and PPL timer information.

L3 Call Reference Configuration Bytes

These values are for connection to the Lucent 4ESS. See L3 Default Settings per Connection Type 8-27 for variations from the defaults shown here for other Connection Types.

Asterisks (*) denote default values.

Byte	Description	Values
0x01	Connect Ack Option (Send a Connect Ack in response to a Connect.)	0x00 = Disabled (False) * 0x01 = Enabled (True)
0x02	Reserved	None
0x03	Status Enquiry Option (Send a Status Inquire message rather than a Status message when receiving an invalid message in the NULL state.)	* 0x00 = Disabled 0x01 = Enabled
0x04	The accepted received Bearer Capability Bit Mask.	0x01 = Voice 0x02 = 3.1 KHz Audio * 0x03 = Voice and 3.1 KHz Audio 0x04 = Unrestricted 56K 0x08 = Unrestricted 64K 0x10 = Restricted 64K 0x20 = 384 K 0x40 = 7.1 KHz Audio
0x05	Channel ID encoding of extension bit (octet 5 in channel ID IE)	* 0x00 = Extension (Ext.) bit set 0x01 = Ext. bit cleared
0x06	Call Proceeding Option (send call proceeding once the call is answered, not automatically after receiving SETUP)	* 0x00 = Disabled 0x01 = Enabled
0x07	Number of SETUP resends (after T303 expiry) NOTE: Not applicable for JATE variant	0x01 = Resend

Byte	Description	Values
0x08	Acceptable Information transfer rate. In BCC IE for incoming calls.	0x01 * = CIRC (Circuit) mode 64K 0x02 = CIRC mode 2_64K 0x04 = CIRC mode 384K 0x08 = CIRC mode 1563K 0x10 = CIRC mode 1920K 0x20 = CIRC mode MULTIRATE
0x09	Send CONNECT ACK indication to the host.	0x00 = Disabled (default) 0x01 = Enabled
0x0A	Call Reference Allocation can be set to (FALSE) which always takes the next value in the list, or (TRUE) which takes the first unused value always searching from the start of the list.	0x00 = False (Default) 0x01 = True
0x22	Number of re-transmission of Status Enquiry Request.	*0x00
QSIG/PSS1 only		
0x23	Send Unexpected Segmented Message to host	0x00 = Disabled *0x01 = Enabled
0x24	Send Segmentation Failure - Wrong or Unexpected Segmented Message to host	0x00 = Disabled *0x01 = Enabled
Euro-ISDN Only		
0x08	Connect ACK to host	0x00* = Do Not Send
0x09	STATUS to host	0x00* = Do Not Send
0x0B	Glare Control	0x00* = Drop Outgoing

L3 Call Reference PPL Timers

The table below shows the PPL Timer values for the ISDN component ISDN L3 Call Reference (0x0008).

* denotes use for Euro-ISDN only. ** denotes use for Lucent 4ESS, 5ESS and DMS-100 and 250 only. Refer to *API Reference* for more information about ISDN.

Timer ID	Description	Default Values (seconds)
0x0002*	Q.931 T302	15
0x0003	Q.931 T303	4
0x0004*	Q.931 T304	15
0x0005	Q.931 T305	4
0x0008	Q.931 T308	4
0x000A**	T310**	25
0x000D	Q.931 T313	4

L3 CR NI 2 User Side Timers

The following table shows the PPL Timers available for use with L3 CR NI 2 User Side:

Timer	Timer Index	Value (10 ms)
T303	3	400
T305	5	400
T308	8	400
T309	9	9000
T310	10	3000
T313	13	400
T322	22	400

L3 CR NI 2 Network Side

The following table shows the PPL Timers available for use with L3 CR NI 2 Network Side:

Timer	Timer Index	Value (10 ms)
T301	1	30000
T303	3	400

Timer	Timer Index	Value (10 ms)
T305	5	400
T306	6	3000
T308	8	400
T309	9	9000
T310	10	3000
T313	13	400
T322	22	400

L3 Call Reference PPL Events

The table below shows the PPL Event Indications for QSIG/PSS1.

PPL Event Indication	0x0001
Name	Unexpected Message Segment
Description	The CSP sends this PPL Event Indication to the host when a Segmented Message, coming from the upper layer, is not proceeded by a call processing message containing a Segmented IE. This usually happens when the host sends a L4 SEGMENT PPL Event Request, without first sending a call processing message containing a Segment ICB.
PPL Event Indication	**0x0002**
Name	Segmentation Failure - Wrong or Unexpected Segmented Message
Description	The CSP sends this PPL Event Indication to the host when a failure occurs in the segmentation procedure due to an invalid Segmented Message or Segmented Message with a different Call Reference IE.

L3 Global Call Reference (0x0009)

Purpose This section describes the PPL configuration bytes, events and timers for the L3 Global Call Reference PPL component.

L3 Global Call Reference Configuration Bytes

Byte	Description	Values
0x01	Channel ID encoding of extension bit	* 0x00 = Ext. bit set 0x01 = Ext. bit cleared
0x02	N316: Number of RESTART attempts	0x02
0x03	Send FACILITY INDICATION to the host	* 0x00 = Disabled 0x01 = Enabled
0x04	Send FACILITY ACK INDICATION to the host	* 0x00 = Disabled 0x01 = Enabled
0x05	Send FACILITY REJECTED INDICATION to the host	* 0x00 = Disabled 0x01 = Enabled
0x06	Send RESTART to the host	* 0x00 = Disabled 0x01 = Enabled
0x07	Send RESTART ACK to the host	* 0x00 = Disabled 0x01 = Enabled
0x08	Send STATUS to the host	* 0x00 = Disabled 0x01 = Enabled
0x09	Send STATUS ENQUIRY to the host	* 0x00 = Disabled 0x01 = Enabled
0x0A	Send USER INFORMATION to the host	* 0x00 = Disabled 0x01 = Enabled
	* Default value	

L3 Global Call Reference L3/L5 Events

The sending of the PPL events to the host is configurable with PPL Configuration Bytes 3–10. These events are only valid using the ISDN Raw IE type (0x11).

PPL Event Request (L5 to L3)	PPL Event Indication (L3 to L5)
0x0001 Facility	0x0001 Facility Indication
0x0002 Facility ACK	0x0002 Facility ACK Indication
0x0003 Facility REJ	0x0003 Facility REJ Indication
0x0004 Restart	0x0004 Restart
0x0005 Restart ACK	0x0005 Restart ACK
0x0006 Status	0x0006 Status
0x0007 Status Enq	0x0007 Status Enquiry
0x0008 User Info	0x0008 User Information

L3 Global Call Reference PPL Timers

Timer ID	Description	Value (default)
0x16	T316	120 seconds (12000 10ms units)

L3 D Channel Control (0x000A)

Purpose This section describes the PPL configuration bytes and events for L3 D Channel Control.

L3 D Channel Control Configuration Bytes

Byte	Description	Values
0x01	Channel ID encoding of extension bit	* 0x00 = Ext. bit set 0x01 = Ext. bit cleared
0x02	Network Side (T309 Logic) Enable/ Disable	0x00 = Disable 0x01 = Enable Default is enabled if Connection Type is a Network Side variant.
0x09	Send D channel Up event to GCR	0x00 = Disable *0x01 = Enable

L3 D Channel Control L3/L5 Events

PPL Event Request (L5 to L3)	PPL Event Indication (L3 to L5)
0x0001 Manual D Channel Switchover *	0x0001 D Channel Active * 0x0002 D Channel Standby * 0x0003 D Channel Not Aligned *
	0x0004 -Network Side D Channel Down, T309 started
	0x0005 - T309 expired. Network Side D Channel Down
	0x0006 - Network Side D Channel Re-Established.
* Applies to D channel Backup only.	

PPL Timers DSM 11 is D channel control. DSM 12 is D channel control with a backup D channel.

The following PPL timer is available for use with DSM 11:

Timer	Timer Index	Value (10 ms)
T309	9	9000

The following PPL Timer is available for use with DSM 12:

Timer	Timer Index	Value (10 ms)
T309	9	9000

QSIG/PSS1 PPL Timer

The following PPL timer is available for use with QSIG/PSS1.

Timer ID	Description	Value (default)
0x0E	T314 (Reassembly Timer)	4 seconds (400 10ms units)

L3 B Channel Control (0x000B)

Purpose This section describes the PPL configuration bytes for L3 B Channel Control.

L3 B Channel Control Configuration Bytes

Byte	Description	Values
0x01	Channel ID encoding of extension bit	* 0x00 = Ext. bit set 0x01 = Ext. bit cleared
0x02	Enable support for sub-configured interface	0x00 = Disabled (Default) 0x01 = Enabled
0x03	Sending the indication from BSS to host reporting the B channel state	0x00 - Disable send Indications (Default) 0x01 - Enable Send Indications

Event Request

Event ID	Event
0x07	Bring channel INS
0x08	Busy out B channel

Event Indications

Event ID	Event
0x02	B channel state OOS
0x03	B channel state INS

L4 Call Control Channel Management (0x0061)

Purpose

This section describes the L4 Call Control Channel Management uses PPL Event information for QSIG/PSS1.

L4 Call Control Channel Management QSIG/PSS1 PPL Events

The table below shows the PPL Event Request and Indication for QSIG/PSS1.

PPL Event Request	**0x0010**
Name	ISDN Segment
Description	This PPL Event Request allows the host to send Segment Message to L4CH. The Data ICB, ISDN Segmented Message (0x25) must be included. Also Data ICB Formatted ISDN IEs or Raw ISDN IEs must be included in the *PPL Event Request* API message.
PPL Event Indication	**0x0010**
Name	ISDN Segment
Description	This PPL Event Indication allows the CSP to report to the host when the Segment Message comes from L3. Data ICB ISDN Segmented Message (0x25) will be included. Also, Data ICB Formatted ISDN IEs or Raw ISDN IEs must be included in the *PPL Event Indication* API message.

10 DASS2/DPNSS

Purpose The Dialogic DASS2/DPNSS card (Digital Access Signaling System2/ Digital Private Network Signaling System) provides D channel packet processing and call control procedures.

DASS2/DPNSS Features & Internal Architecture

Overview

This section describes features included with DASS2 and DPNSS, a diagram of the software architecture

DASS2

The DASS2 variant features the following:

- 16 D channels supported per card
- BS 7378 and BTNR 190 compatibility
- Flexible Call Control Management using PPL control components
- User side support

DPNSS

The DPNSS variant features the following:

- 16 D channels each supporting 30 B channels
- BTNR 188 compatibility
- Flexible Raw DPNSS message management
- PBX A and B, X and Y support
- Virtual call capability

Supported D Channels

The Layer 2 specifications for both DASS2 (BTNR 190) and DPNSS (BTNR 188) call for the continuous re–transmission of UI (C) frames. UI (C) frames are the basic transport frames that carry all of the DASS2/DPNSS Layer 3 messages. UI (C) frames are to be re-transmitted until either:

- A UI (R) response frame has been received, or
- The maximum number of re-transmitted UI (C) frames has been exceeded (64 is the recommended maximum) and the maximum re–transmission period has been exceeded (500 milliseconds).

Ideally, the interval between successive re-transmissions will be large enough so that the Dialogic CSP will respond to every UI (C) frame with a UI (R) before it has a chance to be re-transmitted. However, the only requirement for this time interval is that “At least one flag” must separate the successive frames (BTNR 190, Section 3, 4.2.6.2, June 1992). Essentially this means there is no minimum limit for the interval between re-transmissions.

The retransmission requirement will affect the performance of the Dialogic DASS2/DPNSS card with regard to how many links can be supported per card. The degree to which performance is affected

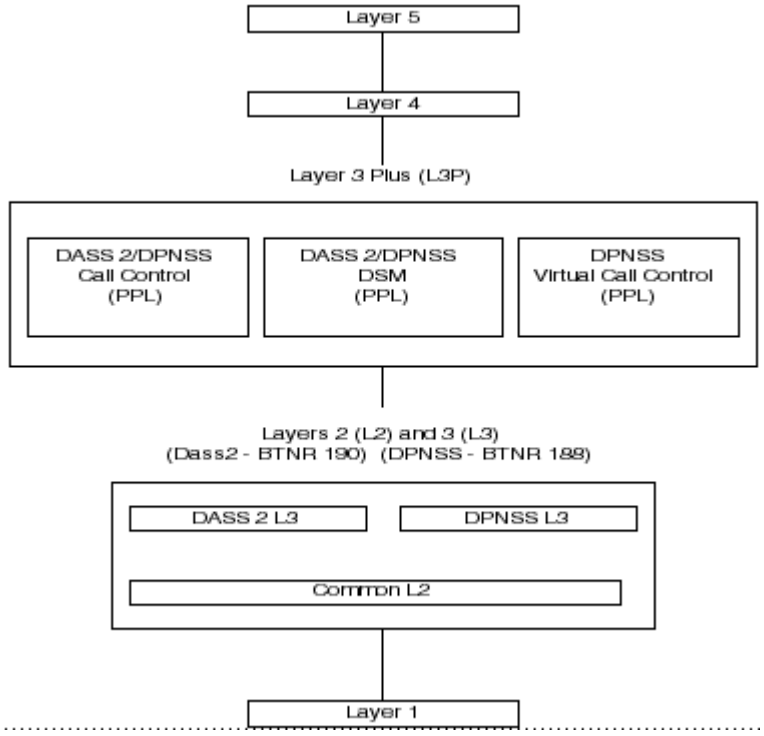
depends on the value of this re–transmission interval as implemented on the PBX or network switch to which the CSP is connected. The shorter this interval, the more UI (C) frames that are re-transmitted to the CSP, and the slower the performance of the DASS2/DPNSS card.

Because the interval between re-transmitted UI (C) frames will vary from PBX to PBX (or from one network switch to another), the performance of the DASS2/DPNSS card, and therefore the number of links supported per card, will also vary.

Architecture

The figure below illustrates the functional modules involved in the implementation of the DASS2/DPNSS card.

Figure 10-1 DASS2/DPNSS High-Level Architecture



More on DASS2/DPNSS modules

L3P

The Layer 3 Plus (L3P) module is an interface between Layer 3 and Layer 4 Central Call Processing (CCP) on the Matrix Controller, and Layer 5 (host). L3P formats information from Layer 3 into the programmed format for the host.

- DASS2 Call Control [Component (0x0D) per B channel]

  This component is the interface between Layer 3 and Layer 4 CCP, which resides on the Matrix Controller. This component manages which message to send internally for call processing and notifies the host of various events related to the protocol.

- DASS2 D Channel Control [Component (0x0E) per D channel]

  This component which is the D channel State Machine (DSM), manages the enabling and disabling of the D channel. It informs the L3P DASS2 Call Control component of the availability of the D channel.

L3 and L2

Layers 2 and 3 are combined into one module on the DASS2/DPNSS card. This module provides an implementation of the Link Access protocol (Level 2) as defined in the BTNR 190, Volume 1, Section 3. This module is also an implementation of Call Control Signaling as defined in BTNR 190, Volume 1, Section 4-8. This module is not PPL controlled.

Configuring DASS2/DPNSS Software

Overview This section describes how to configure the DASS2/DPNSS software.

Before you begin Ensure that the DASS2/DPNSS card is installed correctly.

Configuring DASS2/DPNSS The table below describes the sequence of steps to perform for the system assignment.

Step	Action	Description	Message
1	Assign Spans	Assign all spans to be used for DASS2/DPNSS	Assign Logical Span ID
2	Configure Spans	Configure framing and line coding for all DASS2/DPNSS spans.	E1 Span Configure
3	Assign D Channels	Assign all D channels	D Channel Assign
4	Assign DPNSS as variant	Reconfigure to conform to DPNSS protocol	ISDN Interface Configure
5	Configure PPL	Download and assign custom protocols and perform any timer or Config. Byte modifications required.	See PPL messages
6	Bring All Spans and Channels in Service	Bring spans, B channels, and D channels in service.	Service State Configure

More details on the configuration steps

The following steps describe each action in the table:

1. Assign spans

   Assign Logical Span IDs to spans on SE1LC cards to be used for DASS2/DPNSS with the *Assign Logical Span ID* message.

2. Configure spans

   Set the format **(***E1 Span Configure*) of the span on which the DASS2/DPNSS D channel resides. Configure the framing and line coding for all DASS2/DPNSS spans by using HDB3 and Clear Channel.

3. Assign D Channels

   Assign D channels using the *D Channel Assign* message. The logical channel must be 30. In order for this assignment to succeed, the SE1LC card on which the D channel is to reside must be in service. The physical span must be assigned. You must specify the slot number of the DASS2/DPNSS card, as well as the type of primary.

   The system software selects the actual D channel resource. If the host attempts to assign a D channel to a DASS2/DPNSS card that already has 32 D channels assigned, the host receives a Response Status of "DASS2 D Channel Exceeds Max" (0xA4). You must indicate another card for D channel processing. All subsequent references to the D channel timeslot use the Logical Span ID and channel specified in the message.

Requirements for Assigning a D Channel

- The span must be assigned.
- The SE1LC cards must be present and operational.
- The DASS2/DPNSS card must be operational and have available D channel(s).
- The D channel must reside on channel 30 (timeslot 16).
- See *ISDN D Channel Configuration (8-24)* for example messages of D channel configuration.

4. Assign DPNSS as variant

   Reconfigure the E1 lines from the DASS2 protocol to conform to the DPNSS protocol with the *ISDN Interface Configure* message. This step is necessary only if you are using DPNSS.

Important! The value of the *Entity* field (0x01) indicates that the entity being addressed is the connection type. The value of the *Data[1]* field (0x0c) indicates that the connection type is DPNSS.

5. Configure PPL

   Perform any PPL customization required, including downloading and assigning custom protocols and timer or Config Byte modifications. See *DASS2/DPNSS (10-1)* for more information.

6. Bring Spans and Channels in Service

   When all of the configuration is complete, bring up the D channel and the B channels to establish a connection with the network and then begin call processing. Upon successful establishment, the switch sends the *DS0 Status Change* message to the host with the status of in-service.

   DS0 Status Change messages follow for the B channels (voice/data channels). *PPL Event Indication* messages are sent for each DPNSS virtual channel that comes into service.

   Upon link failure, the switch sends the *DS0 Status Change* message to the host for the D channel, as well as all of the associated B channels. All channels have an “out-of-service” status and *PPL Event Indication* messages are sent for all DPNSS virtual channels.

Call Flows

Overview

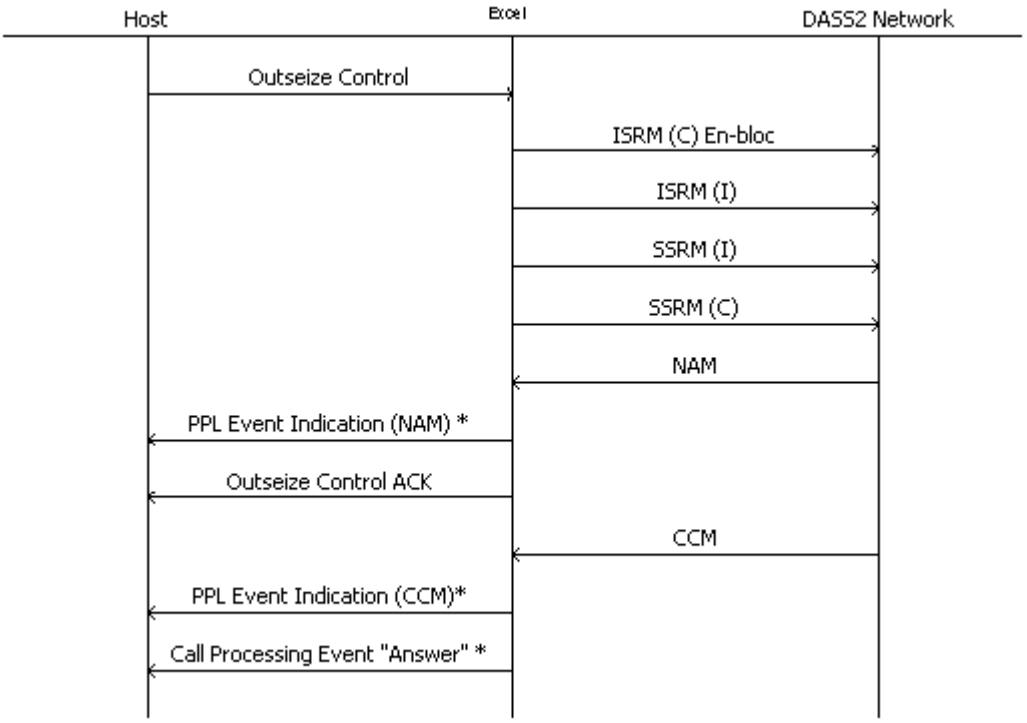
This section describes call flows for DASS2, DPNSS, and DPNSS Virtual Calls.

Many of the following call flows have optional messages based on the configuration. If a message is optional, an asterisk is displayed after the message.

DASS2 Call Flows

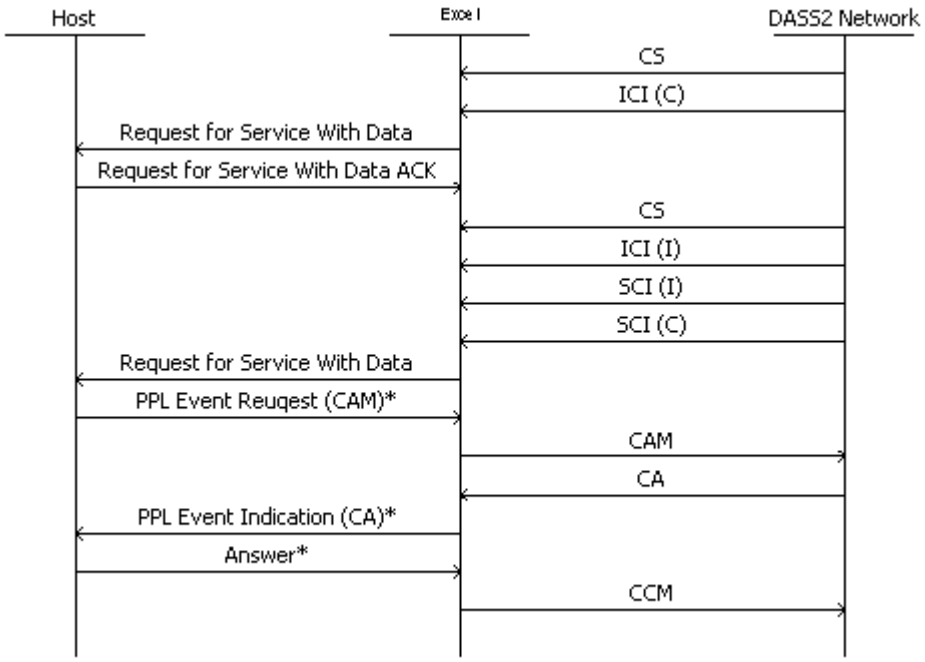
Refer to the following figures for DASS2 call flow information.

DASS2 Outgoing Call



* Optional Message

DASS2 Incoming Call



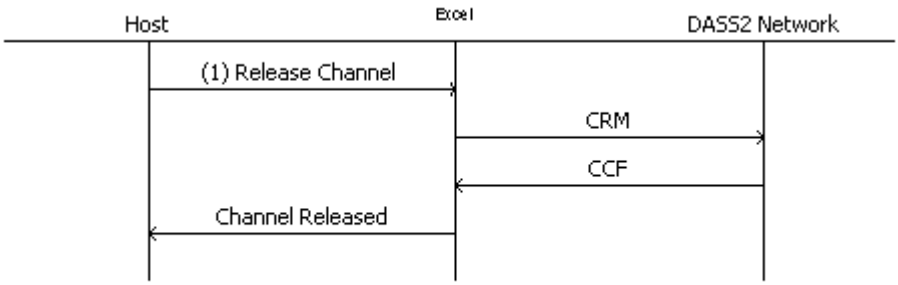
The above call flow shows the sequence for two scenarios for Request for Service With Data.

The Answer is propagated by L4 or from Generate Call Processing Event).

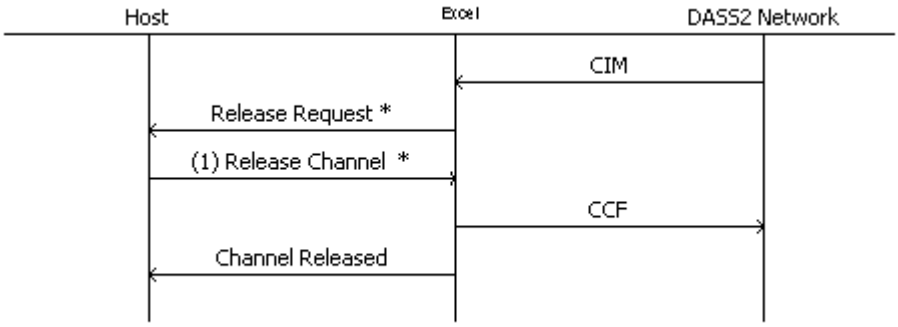
* Optional Message.

DASS2 Call Clearing

User-Initiated Call Clearing



Network-Initiated Call Clearing



(1) Release Channel is Release Channel With Data.

* Optional Message

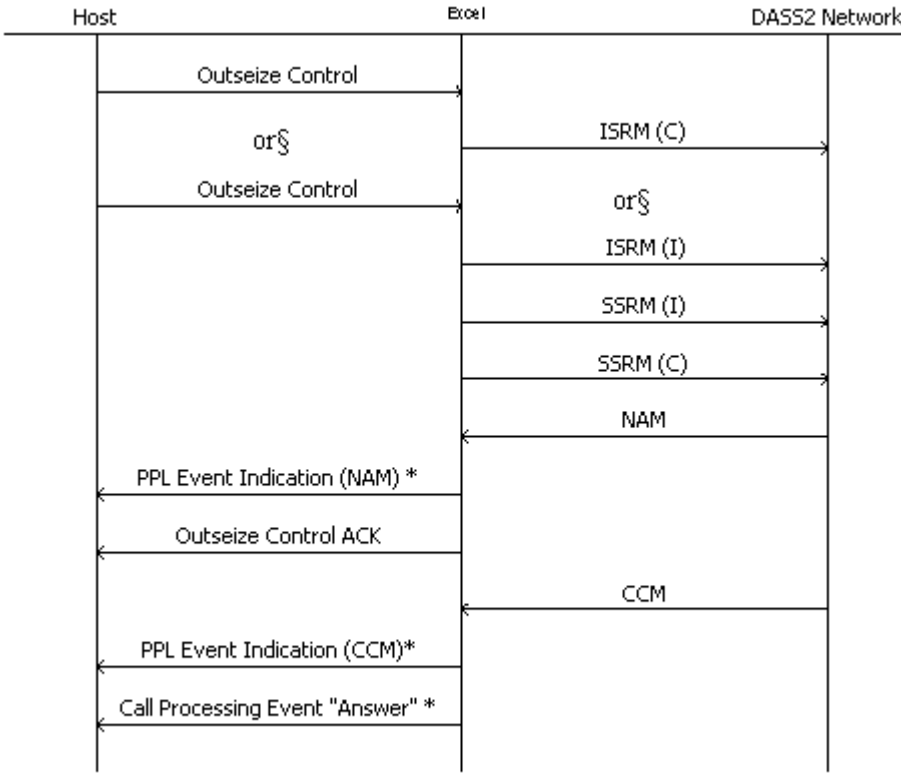
DPNSS Call Flows

The following figures provide DPNSS call flow information.

DPNSS Outgoing Call En-bloc or Overlap Sending Mode

The switch automatically chooses whether to send the call en-bloc or overlap mode. If the size of the data in the ISRM is 45 bytes or less, the switch sends the call out in en-bloc mode. If the size of the data is greater than 45 bytes, the call goes out in overlap mode.

You have the option of forcing an outgoing call in overlap mode. You perform this by setting the value of Configuration Byte 2 to 0x01.



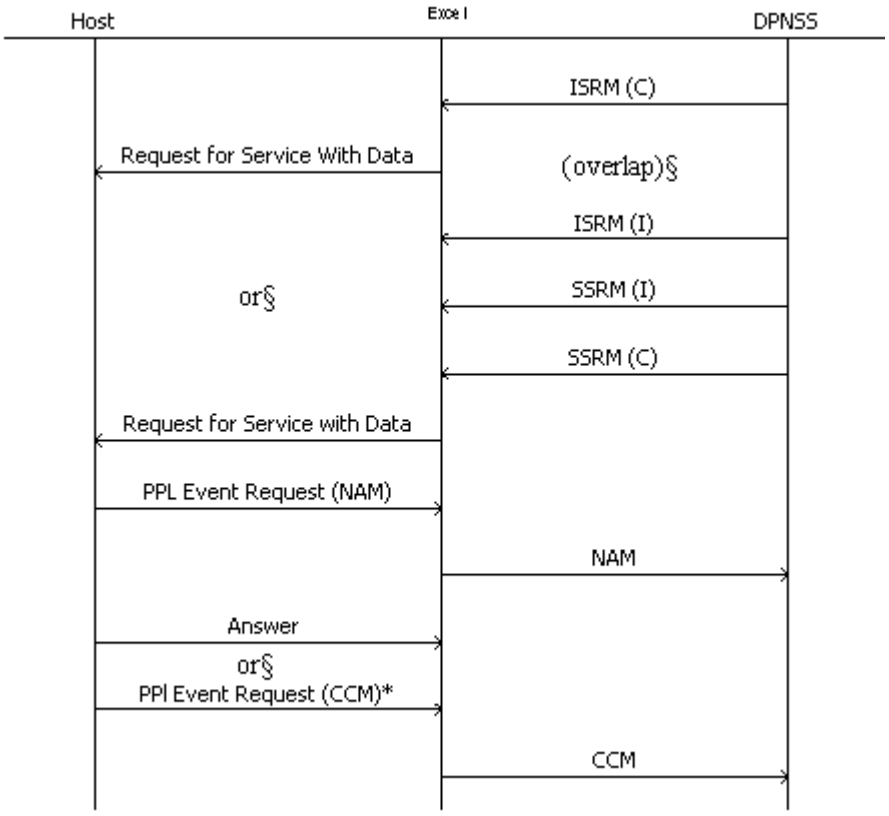
* Optional Message

DPNSS Incoming Call En-bloc Receiving

An ISRM(C) will generate a *Request For Service With Data* message to the host.

DPNSS Incoming Call Overlap Receiving

An ISRM (I) will be followed by a sequence of SSRM(I), concluded by an SSRM(C). The concatenated data from these service request messages will then be passed to the host in *Request For Service With Data*.



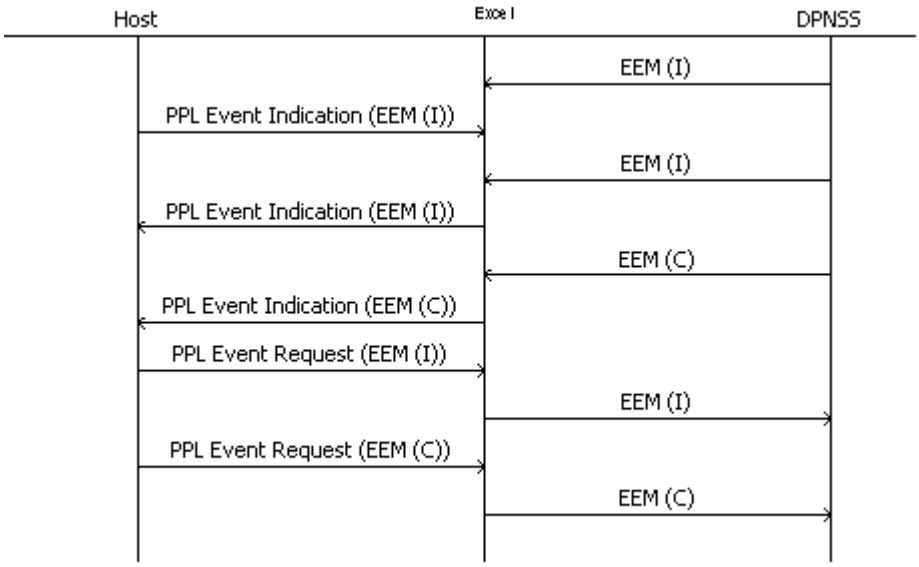
The above call flow shows the sequence for two scenarios for *Request for Service With Data.*

The Answer is propagated by L4 or from Generate Call Processing Event).

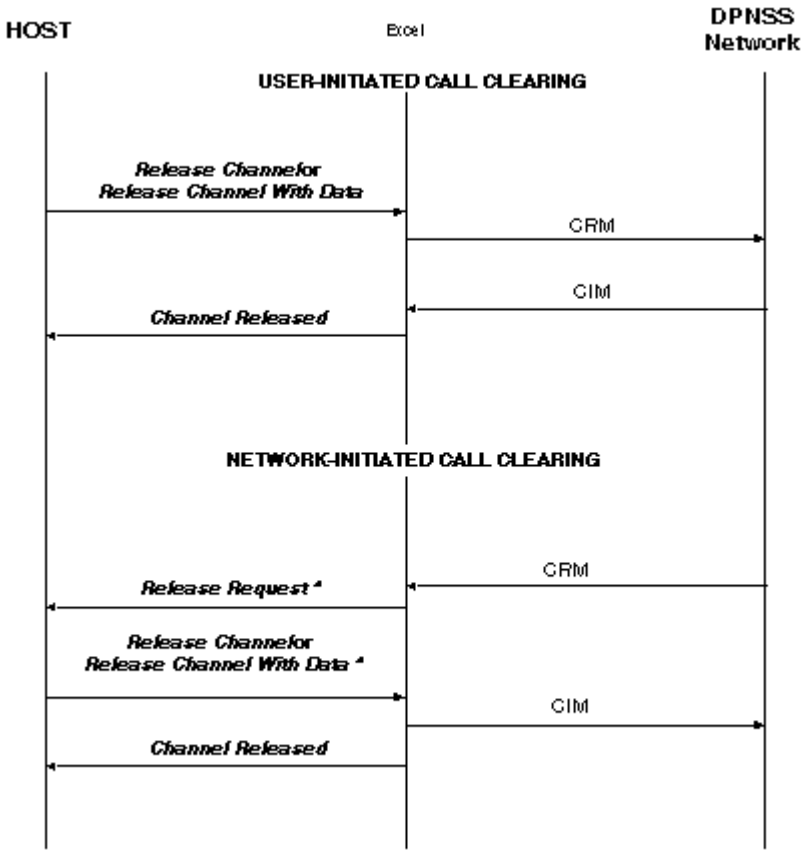
* Optional Message.

DPNSS End-to-End Message (EEM) Complete/Incomplete

Once a call is established, DPNSS supplementary information can be sent or received using the EEM message.



The following call flows depict a Network-Initiated DPNSS Clear Request Message and a Host Initiated DPNSS Clear Request Message.



DPNSS Virtual Call Control

DPNSS supports the capability of having two calls up on the same channel simultaneously—one real and one virtual. Within a virtual call, the call control information is passed but does not use a voice channel.

Consequently, only real calls go through Layer 4 and virtual calls are controlled with direct messaging from the host to L3P via *PPL Event Request* and *PPL Event Indication* messages. Virtual calls are not able to use standard call control API messages since these messages utilize Layer 4. Standard API messages include:

- *Outseize Control*
- *Request for Service with Data*
- *Park Channel*
- *Connection* messages
- *Release, Release with Data*

All API messages between the virtual call PPL component and the host have the PPL component ID 0x51. This component value is the only way of distinguishing whether a message pertains to a real or virtual call.

With all virtual call control being communicated through the PPL Event Request and PPL Event Indication messages, the translation values in the table below must be known.

Important! All component values are 0x51.

Virtual call PPL event values are as follows:

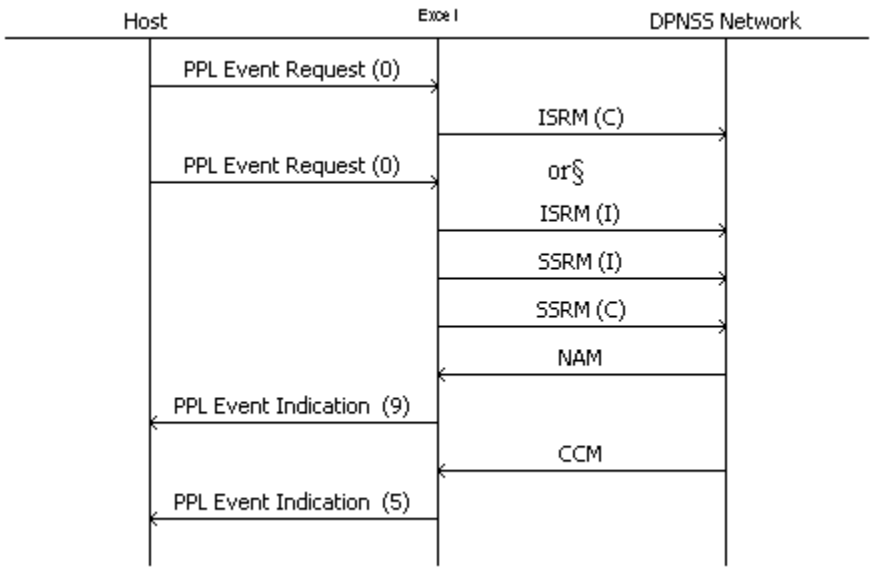
DPNSS Message	PPL Event Value
NAM	0x09
CCM	0x05
CRM	0x08
CRM	
CCF	
ISRM (C)	0x00
ISRM (I)	0x01
EEM (C)	0x22
EEM(I)	0x23
SSRM (C)	0x11
SSRM (I)	0x12

DPNSS Virtual Call Control Call Flows

DPNSS Virtual Outgoing Call En-bloc or Overlap Sending

The switch automatically chooses whether to send the call en-bloc or overlap mode. If the size of the data in the ISRM is 45 bytes or less, the switch sends the call out in en-bloc mode. If the size of the data is greater than 45 bytes, the call goes out in overlap mode.

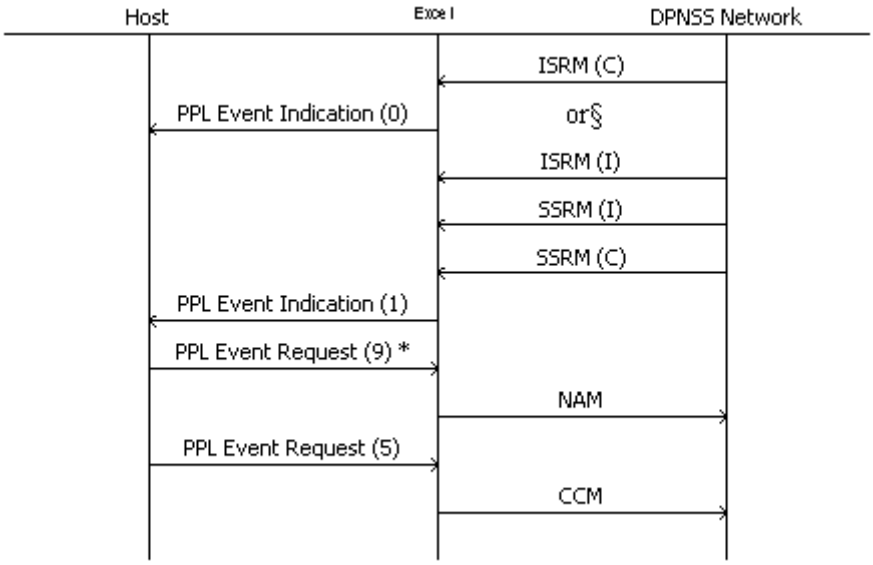
You have the option of forcing an outgoing call in overlap mode. You perform this by setting the value of Configuration Byte 2 to 0x01.



Important! In a DPNSS Incoming Call En-bloc Receiving call flow, an ISRM(C) will generate a *Request For Service with Data* message to the host.

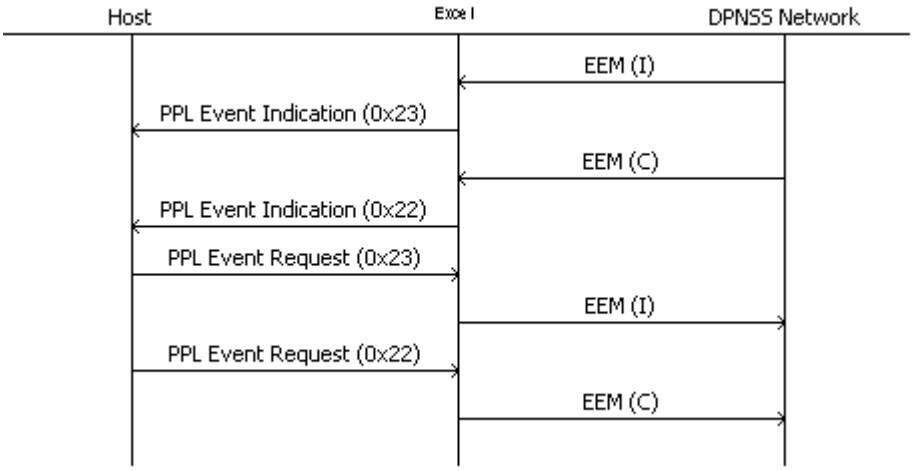
DPNSS Virtual Incoming Call Overlap Receiving

An ISRM (I) will be followed by a sequence of SSRM(I), concluded by an SSRM(C). The concatenated data from these service request messages will then be passed to the host in a *PPL Event Indication* message.



DPNSS Virtual Supplementary Message Call Flow

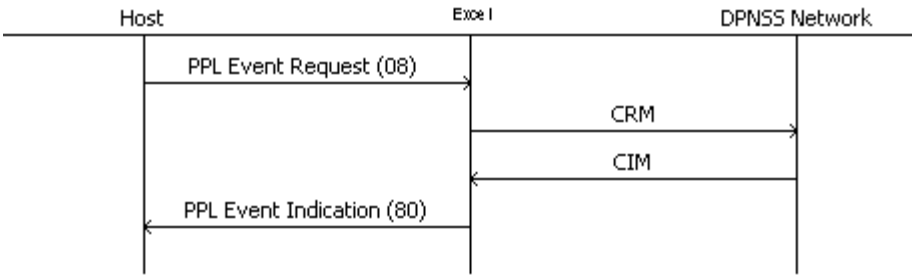
Once a call is established, DPNSS supplementary information can be sent or received using the EEM message.



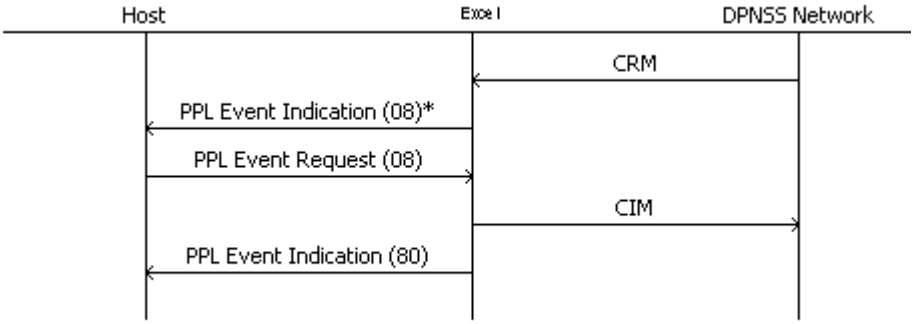
Network-Initiated DPNSS Virtual Call Clear Request

The following call flows depict a User-initiated DPNSS Virtual Call Clear Request Message and then a Network-Initiated DPNSS Virtual Call Clear Request Message.

User-Initiated Call Clearing



Network-Initiated Call Clearing



11 DASS2/DPNSS PPL Information

Purpose This chapter provides PPL information for DASS2/DPNSS

For a list of the PPL Components and addressing information, see *PPL Component IDs* and *PPL Component Addressing* in the *API Reference*.

DASS2/DPNSS L3P Call Control (0x0D)

Overview

This section includes the PPL configuration bytes and events for DASS2/DPNSS L3P Call Control

L3P Call Control L3P/L5 Events

The following two tables show the L3P Call Control L3P/L5 Events supported for DASS2 and DPNSS.

DASS2

The table below shows the DASS2 L3P Call Control L3P/L5 Events that are supported by Dialogic. The table includes the direction in which the message is being sent between the public branch exchange (PBX) and the exchange termination (ET).

Message	Description	PPL Event	Direction
ISRM (C)	Initial Service Request Message (Complete)	0x00	PBX→ET
ISRM (I)	Initial Service Request Message (Incomplete)	0x01	PBX→ET
SSRM (I)	Subsequent Service Request Message (Incomplete)	0x0B	PBX→ET
SSRM (C)	Subsequent Service Request Message (Complete)	0x0C	PBX→ET
CS	Channel Seized	0x04	ET→PBX
ICI (C)	Incoming Call Indication (Complete)	0x00	ET→PBX
ICI (I)	Incoming Call Indication (Incomplete)	0x01	ET→PBX
SCI (I)	Subsequent Call Indication (Incomplete)	0x0B	ET→PBX
SCI (C)	Subsequent Call Indication (Complete)	0x0C	ET→PBX
CAM	Call Accepted	0x09	PBX→ET
CA	Call Arrival	0x07	ET→PBX
NAM	Number Acknowledge Message	0x09	ET→PBX
CCM	Call Connected Message	0x05	Both
CRM	Clear Request Message	0x08	PBX→ET
CCF	Clear Confirmation Message	0x08	PBX→ET

Message	Description	PPL Event	Direction
CIM	Clear Indication Message	0x08	ET→PBX
SM	Swap Message	0x20	Both
MIM (C)	Maintenance Information Message	0x43	Both

The following DASS2 messages are not supported by Dialogic:

- NIM
- RM (C)
- RRM
- SRW
- SeSR
- UUD(C)
- UUD(I)
- UDC

DPNSS

The table below shows the DPNSS L3P Call Control L3P/L5 Events that are supported by Dialogic.

Message	Description	PPL Event
CCM	Call Connected Message	0x05
CIM	Clear Indication Message	0x08
CRM	Clear Request Message	0x08
EEM(C)	End-to-End Message (Complete)	0x22
EEM(I)	End-to-End Message (Incomplete)	0x23
ISRM(C)	Initial Service Request Message (Complete	0x00
ISRM(I)	Initial Service Request Message (Incomplete)	0x01
NAM	Number Acknowledge Message	0x09
SM	Swap Message	0x44

Message	Description	PPL Event
SSRM(C)	Subsequent Service Request Message (Complete)	0x0C
SSRM(I)	Subsequent Service Request Message	0x0B

The following DPNSS messages are not supported by Dialogic:

ERM(C)	NIM
ERM(I)	RM(C)
LLM(C)	RM(I)
LLM(I)	RRM
LLRM	SCIM
LMM	SCRM
LMRM	

DASS2/DPNSS L3P Call Control Configuration Bytes

The PPL Config bytes for the DASS2/DPNSS L3P CC component vary depending on whether you are using the DASS2 or DPNSS connection type.

DASS2

Byte	Description	Values
0x01	Request for Service Format	* 0x00 = BCD Encoded 0x01 = Raw ICB
0x02	Send in either En-bloc or overlap mode	* 0x00 = En-bloc/overlap 0x01 = Overlap always
0x03	Aculab Receive Mode	* 0x00 = Disable 0x01 = Enable
0x04	Reject SM, do not send to host; or accept SM, send SM to host, and allow host to respond with SM Request	* 0x00 = Reject 0x01 = Accept

Byte	Description	Values
0x05	Incoming valid SIC bit mask: Bit 0: Speech A-law 64 Kb/s Telephony (SIC 00) Bit 1: Speech A-law 64 Kb/s Category 2 (SIC 10) Bit 2: Speech A-law 64 Kb/s Category 1 (SIC 12) Bit 3: Data 64 Kb/s (SIC A0) Bit 4: 3.1 KHz Audio (SIC 18) Bit 5: 3.1 KHz Audio with group 2 & 3 Fax (SIC 1F) Bit 6: Unused Bit 7: Disable incoming SIC validation	* 0x3F = Enable Bits 0-5
0x06	Reserved	
0x07	Reserved	
0x08	Reserved	
0x09	Outgoing SIC, Octet 1 (BCD type Outseize Control only)	* 0x00
0x0A	Outgoing SIC, Octet 2 (BCD type Outseize Control only)	* 0x00
0x0B	Send received NAM to host	* 0x00 = Disable 0x01 = Enable
0x0C	Send CIM/CRM to host	* 0x00 = Disable 0x01 = Enable
0x0D	Send CA to host	* 0x00 = Disable 0x01 = Enable
0x0E	Send ICI(I) to host	* 0x00 = Disable 0x01 = Enable
0x0F	Send SCI(C) to host	* 0x00 = Disable 0x01 = Enable
0x10	Send SCI(I) to host	* 0x00 = Disable 0x01 = Enable
0x11	Send CCF to host	* 0x00 = Disable 0x01 = Enable
0x12	Send CCM to host	* 0x00 = Disable 0x01 = Enable
0x13	Reserved	
0x14	Send NAM automatically to distant end upon an incoming call	* 0x00 = Enable 0x01 = Disable

Byte	Description	Values
0x15	Send ICI(C) to host	* 0x00 = Disable 0x01 = Enable

DPNSS

Byte	Description	Values
0x01	Request for Service Format	* 0x00 = BCD Encoded 0x01 = Raw ICB
0x02	Send in either En-bloc or overlap mode	* 0x00 = En-bloc/overlap 0x01 = Overlap always
0x03	Aculab Receive Mode	* 0x00 = Disable 0x01 = Enable
0x04	Reject SM, do not send to host; or accept SM, send SM to host, and allow host to respond with SM Request	* 0x00 = Reject 0x01 = Accept
0x05	Incoming valid SIC bit mask: Bit 0: Speech A-law Bit 1: Data 64 Kb/s Bits 2–7: Unused	* 0x03 = Enable Bits 0 & 1
0x06–0x08	Reserved	
0x09	Outgoing SIC, Octet 1 (BCD type Outseize Control only)	* 0x10
0x0A	Outgoing SIC, Octet 2 (BCD type Outseize Control only)	* 0x00
0x0B	Send received NAM to host	* 0x00 = Disable 0x01 = Enable
0x0C	Send CIM/CRM to host	* 0x00 = Disable 0x01 = Enable
0x0D	Reserved	
0x0E	Send ISRM(I) to host	* 0x00 = Disable 0x01 = Enable
0x0F	Send SSRM(C) to host	* 0x00 = Disable 0x01 = Enable

Byte	Description	Values
0x10	Send SSRM(I) to host	* 0x00 = Disable 0x01 = Enable
0x11	Send CCF to host	* 0x00 = Disable 0x01 = Enable
0x12	Send CCM to host	* 0x00 = Disable 0x01 = Enable
0x13	Reserved	
0x14	Send NAM automatically to distant end upon an incoming call	* 0x00 = Enable 0x01 = Disable
0x15	Send ISRM(C) to host	* 0x00 = Disable 0x01 = Enable
0x16	Reserved	
0x17	Send CIM (DPNSS Only)	0x00 = Send with default Cause Code (0x30) * 0x01 = Send with received CRM data.
0x64	DPNSS Layer 3	0x00 for PBX (X)* 0x01 for PBX (Y)
0x65	DPNSS Layer 2	0x00 for PBX (A)* 0x01 for PBX (B)

DASS2/DPNSS L3P Events

See Common Atomic Functions in the CAS Developer's Guide.

Event ID	Event
0x0A	CFG in Service
0x0B	CFG Out of Service
0x0C	L1 Alive
0x0D	L1 Dead
0x0E	Channel Aligned
0x0F	Channel Not Aligned
0x10	Channel Reset
0x13	CR Purge Indication

Event ID	Event
0x28	L3 ICI(I)/ISRM(I) Indication
0x29	L3 ICI(I)/ISRM(C) Indication
0x2A	Call Accepted Indication
0x2B	Call Arrival Indication
0x2C	L3 CAM/NAM Indication
0x2D	L3 CCM Indication
0x2E	L3 CIM/CRM Indication
0x2F	L3 CCF Indication
0x30	Call Rejection Indication
0x31	L3 SM Indication
0x32	L3 SCI(I)/SSRM(I) Indication
0x33	L3 SCI(C)/SSRM(C) Indication
0x34	L3 EEM(I) Indication (DPNSS only)
End-to-End Incomplete message received from the Network. The Indication data from the EEM(I) may be loaded into a data ICB using atomic function 72 and then sent to the host in a PPL Event Indication using atomic function 64. (DPNSS only.)	
0x35	L3 EEM(C) Indication (DPNSS only)
End-to-End Complete message received from the Network. The Indication data from the EEM(C) may be loaded into a data ICB using atomic function 72 and then sent to the host in a PPL Event Indication using atomic function 64.	
0x50	L4 Call Request
0x51	L4 Alerting Request
0x52	L4 Connect Request
0x53	L4 Internal Connect Request
0x54	L4 Clear Request
0x56	L4 Outseize Control

Event ID	Event
0x57	L4 Purge

DASS2/DPNSS Virtual Call Control (0x51)

DASS2/DPNSS Virtual Call Control Configuration Bytes

Byte	Description	Values
0x01	Reserved	
0x02	Send in either En-bloc or overlap mode	* 0x00 = En-bloc/overlap 0x01 = Overlap always
0x03	Aculab Receive Mode	* 0x00 = Disable 0x01 = Enable
0x04	Reject SM, do not send to host; or accept SM, send SM to host, and allow host to respond with SM Request	* 0x00 = Reject 0x01 = Accept
0x05	Incoming valid SIC bit mask: Bit 0: Speech A-law Bit 1: Data 64 Kb/s Bits 2–7: Unused	* 0x03 = Enable Bits 0 & 1
0x06–0x0A	Reserved	
0x0B	Send received NAM to host	0x00 = Disable * 0x01 = Enable
0x0C	Send CIM/CRM to host	0x00 = Disable * 0x01 = Enable
0x0D	Reserved	
0x0E	Send ISRM(I) to host	0x00 = Disable * 0x01 = Enable
0x0F	Send SSRM(C) to host	0x00 = Disable * 0x01 = Enable
0x10	Send SSRM(I) to host	0x00 = Disable * 0x01 = Enable
0x11	Send CCF to host	0x00 = Disable * 0x01 = Enable
0x12	Send CCM to host	0x00 = Disable * 0x01 = Enable
0x13	Reserved	

Byte	Description	Values
0x14	Send NAM automatically to distant end upon an incoming call	* 0x00 = Enable 0x01 = Disable
0x15	Send ISRM(C) to host	* 0x00 = Disable 0x01 = Enable
0x16	Send CIM automatically to distant end upon receipt of CRM	0x00 = Disable * 0x01 = Enable
0x17	Send CIM (DPNSS Only)	0x00 = Send with default Cause Code (0x30) * 0x01 = Send with received CRM data.

12 V5.2 Protocol

Purpose This chapter describes the Dialogic implementation and features of the V5.2 protocol.

Important! This release of V5.2 supports LE only.

Important! AN is not supported for V5.2. Any text or illustrations (for example, call flows) indicating the AN side is for clarification purposes only.

Introduction

Overview

This section describes Dialogic's V5.2 products, including the hardware and software required to run and manage V5.2 on the Converged Services Platform (CSP). This implementation supports only the Local Exchange (LE) side of V5.2.

This section has been created to help application developers correctly implement and manage V5.2 applications on the CSP. It describes the internal components of the V5.2 implementation that accommodate signaling, configuration, maintenance, startup procedures, and call processing.

The V5.2 implementation involves the E-ONE card and the ISDN Series 3 card, where the V5.2 protocol resides and functions, and where all of the signaling takes place. The C Channel information is internally routed to the ISDN Series 3 card from the E-ONE card.

Description

V5.2 is a concentration protocol for digital Common Channel Signaling (CCS). It is called a concentration protocol because it can accommodate more subscribers than existing physical ports. The V5.2 protocol is comprised of the LE (Local Exchange) side of the protocol. The V5.2 protocol is used to establish, maintain, and release calls between an LE and an AN.

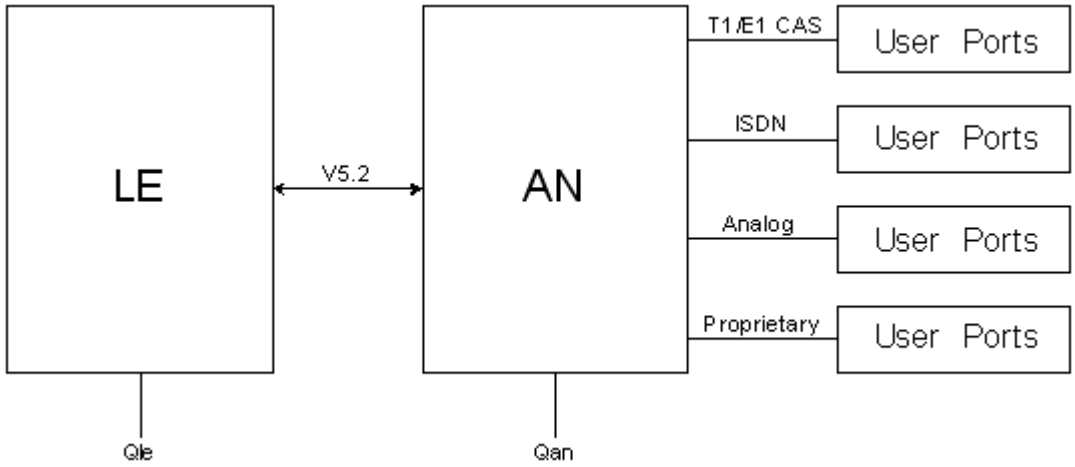
The V5.2 LE is the "network side" that resides at an End Office (EO) and provides the EO and Mobile Switching Center (MSC) functionality. This functionality includes additional CLASS features and services typically provided by an LE. The AN receives V5.2 data from the LE, then passes it to two-wire analog, ISDN, Frame Relay, Wireless, Fiber Optic, or any other Customer Premises Equipment (CPE).

V5.2 supports up to 16 E1s (2048 kbps) per interface. Residing on two of these links are primary and secondary C Channels (the equivalent of ISDN D Channels) and voice/data circuits. The C Channels are 64 kbps HDLC controllers that carry signaling information for various sub-protocols within V5.2. These sub-protocols manage call processing, startup, synchronization, maintenance, and C Channel protection. V5.2 has built-in link protection that permits calls to proceed when the primary C Channel link has failed. V5.2 supports PSTN and ISDN user ports. To maintain service when multiple V5.2 links are available and

one link is lost, bearer timeslots are allocated dynamically. Even if individual calls are lost, they can be re-established on a different link when re-dialed.

Diagram The figure below shows the V5.2 protocol LE relationship.

Figure 12-1 V5.2 Protocol LE Relationship



Supported Features

- V5.2 for the LE side
- 16 interfaces per ISDN Series 3 card
- 16 links (E1) per interface
- 10,000 subscribers (user ports) per ISDN Series 3 card
- PSTN protocol, BCC protocol, Control protocol, Link protocol, Protection protocol, System Management, LAPV, and Envelope function
- Protection Group 1 C Channel pair
- PPL programmability for call processing and layer management
- Dynamic addition and removal of links (not allowed for primary and secondary C Channel links)
- Maintenance support for link blocking and unblocking

- The ISDN Series 3 card enables the DSP-ONE card to collect digits and play out tones.

V5.2 Redundancy

V.52 redundancy is achieved through card-level redundancy on the ISDN Series 3 card. Refer to *ISDN Redundancy (8-69)* to configure.

Summary

Below are the V5.2 Application Program Interface (API) messages, Address Information Blocks (AIB), and Information Control Blocks (ICB). Also listed are modified API messages used to implement V5.2 on the CSP. For detailed information on API messages, AIBs, and ICBs please refer to the *API Reference*.

API Messages for V5.2

- *V5 Configure* (0x7B)
- *V5 Configuration Query* (0x7C)

Existing Messages for V5.2 Configuration

- *PPL Timer Configure* (0xCF)
- *PPL Configure* (0xD7)
- *PPL Assign* (0xD1)
- *PPL Create* (0xD4)
- *PPL Table Initiate Download* (0xD5)
- *PPL Table Download* (0xD6)
- *PPL Delete* (0xDA)
- *PPL Audit Configure* (0xDC)
- *Service State Configure* (0x0A)

Existing Messages for V5.2 Queries

- *PPL Data Query* (0xDE)
- *PPL Audit Query* (0xDD)
- *PPL Protocol Query* (0xDF)

Address Information Blocks (AIBs) for V5.2

- V5 ID (0x2C)
- V5 ID, Link ID (0x2D)
- V5 Interface ID, User Port (0x2F)

Information Control Blocks (ICBs) for V5.2

- V5 Subscriber ID (0x28)

- V5 Formatted IEs (0x29)
- V5 User Port Range (0x2C)
- V5 Statistics Query Data (0x2F)
- V5 Statistics Data (0x30)
- V5 Status Indication (0x31)

V5.2 Licensing

Licensing of V5.2

The licensing of V5.2 is based on the CSP chassis serial number.

Activating Licensing

To activate V5.2, you must use a Product License Key, which Dialogic provides when you purchase the Product License. The key is unique and encrypted.

For details, refer to *Downloading License Keys to the CSP* in the *Licensing Overview* chapter in the *Developer's Guide: Overview* and the *Product License Download* message (0x0079) in the *API Reference.*

You must download the license key each time new system software is downloaded or when you power cycle.

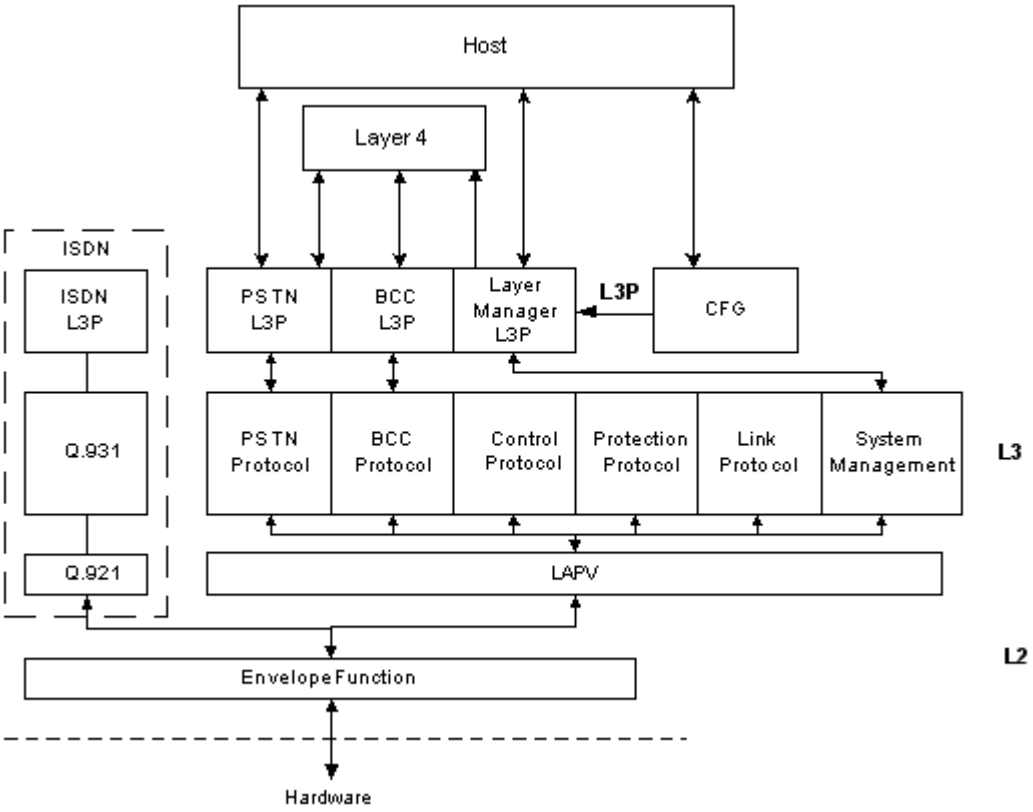
V5.2 Software Architecture

Overview

To gain maximum flexibility for call processing and system management, all three modules at Layer 3 Plus are managed by PPL. This layer provides the interface for Layer 4 call processing for protocol transparency and customer manipulation. The Layer 3 protocols are implemented per specification, while Layer 3 Plus is an additional management layer that enables customer flexibility in case of variant changes.

The figure below shows all of the functional software modules and PPL components in this V5.2 implementation. The host is external to the CSP. Layer 4 resides on the CSP Matrix Series 3 Card. All other modules shown reside on the V5.2 card. The ISDN block (dotted lines) represent future additions to the architecture, and our description pertains to V5.2 related modules only.

Figure 12-2 Software Modules and PPL Components of the V5 Stack



The entire architecture is based upon management of V5 IDs/user ports (the combination uniquely identifies a subscriber), links (where the voice and data traffic are carried) and C Channels (where signaling and control messages are carried). To accommodate all subscribers at once, the host must be aware of the additional requirements for implementing subscribers that outnumber the available physical timeslots. The host's job is simplified because our Layer 4 resource group management handles allocation of available timeslots.

The envelope function is a thin layer closest to the HDLC drivers. It routes incoming packets to the appropriate upper layer task, based upon the envelope address received. Currently this is for LAPV only. LAPV is responsible for guaranteed message delivery, flow control, and sequenced delivery of datagrams to and from the Layer 3 protocols. LAPV is very similar to LAPD; most of the difference exists in the addressing header and format. For one interface, a data link exists for all of the protocols at Layer 3. For interfaces with one link only, there are no protection data links. For interfaces with more than one link, there can be two protection data links.

Each of the protocols that exist at Layer 3 have a unique Layer 3 address. This address allows LAPV to know the protocol for which the messages are destined. In general, they all act as message encoders and decoders, with a state machine for each object they manage. But for system-related events, the System Management has most of the intelligence, and it calls on the appropriate state machines to carry out procedures. For call processing, "call management" is performed at Layer 3 Plus (L3P).

PSTN Protocol The PSTN protocol establishes, clears and restarts PSTN. This call processing state machine manages the call state of the PSTN subscriber. It translates all analog events, provided by a higher layer, and encodes them in a packet, using information elements.

BCC Protocol The BCC protocol manages real time allocation and deallocation of the physical timeslots, which is performed during call processing as the PSTN (or future ISDN) calls demand it.

Control Protocol The Control protocol sends and receives the startup messages. It is also responsible for user port blocking and unblocking.

Protection Protocol

The Protection protocol monitors and maintains the C Channels, so that in case of failure, a switchover can occur to keep signaling traffic flowing on an alternate link.

Link Protocol

The Link protocol tracks the link state and manages the link identification procedures. It also performs link blocking and unblocking.

System Management

System Management coordinates all activities of the Layer 3 protocols. The Finite State Machines (FSM) exist there to manage startup procedures, link activation, restart procedure, accelerated port alignment procedure, and C Channel switchover.

PSTN at L3P communicates to Layer 4 and to the Layer 3 PSTN protocol. The BCC does the same to Layer 3's BCC. Together, these four modules control call processing for the V5 protocol. You can use PPL to modify L3P PSTN and BCC.

The Layer Manager L3P is a PPL Component for controlling the System Management at Layer 3. It has a PPL Event Indication/Request interface to the host for requesting and reporting events on the C Channel status, link states, user port states, statistics querying, etc.

The configuration module is used to manage all configuration and card level aspects. The *V5 Configure* and *V5 Configuration Query* messages are directed there internally. The configuration module is where the configuration database resides.

Configuring V5.2

Overview This section describes how to configure the V5.2 protocol.

All configuration for V5.2 is based on logical V5 IDs. The host uses these logical IDs to manage the physical V5 interfaces in the CSP node. The host manages the configuration of V5 IDs, links, C Channels, logical span IDs, and user ports. V5 IDs contain associated links, C Channels, and user ports. Each V5 ID can contain up to 16 logical link IDs. These link IDs carry all C Channels and voice data. Links are configured per V5 ID, and can exist with or without C Channels.

C Channels are configured per V5 ID, to timeslot 16 on a link. C Channels carry the signaling messages and framing information between two nodes in the V5 network. The host configures a group of user ports for each V5 ID. These user ports are the actual subscribers in the network. Once the valid configuration is complete, the host can bring each V5 interface into service.

The *V5 Configure* API message handles most of the V5 configuration. All queries are configurable using the *V5 Configuration Query* message. When you configure the V5.2 card, the first API message sent to the switch must be the *V5 Configure* message, which creates the V5 interface. The *V5 Configure* message must contain a Slot Address Information Block (AIB) and a *V5 Create* Tag/Length/Value (TLV) block. Then the data portion of the *V5 Create* TLV block contains the V5 ID, and all subsequent configuration and call processing messages address the newly-created V5 ID.

> **Important!** The V5 ID is not the same as the “Interface ID” defined in the ITU-T and ETSI recommendations for V5.2. It is a unique number used by the switch to address the V5 signaling arrangement.

The following entities are required to use the V5.2 protocol in the CSP:

V5 IDs

V5 IDs are the logical representation of physical V5 interfaces. The host creates, configures, and destroys V5 IDs. The host can configure a maximum of 16 V5 IDs per CSP node.

Links

The host can configure a maximum of 16 links per V5 ID.

C Channels

Two C Channels are allowed for each V5 ID.

User Ports

Each ISDN Series 3 card can accommodate a maximum of 10,000 user ports, and a single V5 ID can be configured to use all 10,000 ports.

Logical Span IDs

When adding links to a V5 ID, each Logical Span ID is associated with a Link ID. One link corresponds to one Logical Span ID. The same Logical Span ID cannot be used for more than one V5 interface.

Configuration Sequence

The table below shows a typical sequence for configuring the CSP for V5.2. For more information, see the *V5 Configure* message in the *API Reference*. See the Appendix for a sample configuration trace.

> **Important!** In Step 4, the V5 ID Address Type (AIB) is used to configure all of the options on the V5. The options in step 4 can be performed with a single *V5 Configure* message, using the V5 ID AIB.

Table 12-1 Typical Sequence Configuring CSP for V5.2

Step	Action	API Message
1	Assign Logical Span IDs to the physical E1 spans	*Assign Logical Span ID* (0xA8)
2	Format the E1 spans to clear channel	*E1 Span Configure* (0xD8)
3	Create V5 IDs (using Slot AIB)	*V5 Configure* (0x7B)

Step	Action	API Message
4	Configure V5 Options (using V5 ID AIB): V5 Interface ID Variant ID Variant Type Network Side (LE)	*V5 Configure* (0x7B)
5	Add user port ranges to each V5 ID (using V5 ID AIB)	*V5 Configure* (0x7B)
6	Add links to each V5 ID (using V5 ID AIB)	*V5 Configure* (0x7B)
7	Add C Channels to each V5 ID (using V5 ID AIB)*	*V5 Configure* (0x7B)
8	Bring spans into service	*Service State Configure* (0x0A)
9	Bring V5 IDs into service	*Service State Configure* (0x0A)

*Protection Group 1 is supported for the C Channel type. The physical channel is timeslot 16. Protection Group 2 is not supported.

Important! When V5.2 is configured, the time taken before the newly activated matrix can process calls is longer than with non-V5.2 protocols running. The time taken will be dependent on the details of the V5.2 configuration.

V5.2 Maintenance and Administration

Overview

Normally, the CSP maintains the V5 interface after configuration. By default, it notifies the host upon failures. But you must still monitor the interface status, link status, and user port status to assure the healthy operation of the V5 arrangement.

When performing maintenance on your V5 system, you may require V5 interface, link, and user port management.

V5 Interface

The host can send a *PPL Event Request* for a V5 Interface state to be reported back to the host in a *PPL Event Indication*. This *PPL Event Indication* returns if there is an available communication to the far end. The PPL software also includes a PPL Event Request named *C Channel Status Request* that returns an indication of which C Channel is active.

Link Status

Link status can be queried with the *PPL Event Request* message. The status of all 16 links are returned in individual bytes. The status is a bit mask representation of:

State	Description
0	Link non-operational
1	Link operational
3	Remotely blocked
5	Locally blocked

Link blocking is the last resort blocking mechanism—it should be used only in case of internal failure or emergency.

The LE can reject non-deferred blocking and deferred blocking from the AN side, but it must accept forced blocking. Note that forced blocking can bring the interface out of service if the primary or secondary link is blocked. The LE can block a link, but only forcibly. This forced block automatically tears down any active call and puts each circuit on the link into a blocked state.

The AN side does not share this limitation. Deferred blocking can be requested from the AN side to the LE side, so that the LE side waits until all calls on the link are idle before it blocks the link. This is the

preferred case for non-AN-initiated link blocking. Non-deferred blocking can be requested from the AN side to the LE side, so that the LE side releases all calls, then returns a link blocked indication to the AN side. Our implementation of V5 has been designed so that when a link is blocked, LAPV processing is disabled but Layer 1 is unaffected, and remains aligned. If the host wants to disable Layer 1, it can bring the spans out of service.

User Ports

A user port ID uniquely identifies a subscriber on the V5 interface. In some circumstances, it might be necessary to block or unblock a user port. Blocking and unblocking is accomplished by the host sending PPL Event Requests to the L3P Manager component. When the user port blocking procedures are received, the active call (if there is one) is cleared. Both the AN side and LE side support the blocking and unblocking of the user port. The AN side can also request deferred blocking on a user port, where the user port remains unblocked until the active call goes idle. Our current implementation sends an unblock request if there is a call active, and a block request if it is inactive.

Matrix Switchover

If a Matrix Series 3 card switchover occurs when V5.2 interfaces are in service, the time taken before the newly activated CSP Matrix Series 3 Card can process calls is longer than with non-V5.2 protocols running. The time taken will be dependant on the details of the V5.2 configuration.

V5.2 Startup

Purpose This section describes the automated V5.2 Startup Types.

Startup Sequences The V5 startup procedures vary, depending upon what the LE side and AN side support for port alignment procedures and restart procedures. Two startup sequences are shown below:

- Typical Startup Sequence
- Typical Startup Sequence with Accelerated Port Alignment

1 Refer to the desired startup sequence diagram on the next page.

2 Initiate the request for the variant ID and interface ID.

> **Important!** This request has a unique control element ID and is used in Common Control messages. The purpose is to verify that the LE side and AN side are communicating about the same interface and running the same variant. The variant is a number in the range 0-127, that the LE side and AN side agree upon at provisioning.

3 Continue with the startup sequence, if the request for the variant ID and interface ID is successful.

The the flow diagrams below represent typical startup sequences. They do not show the actual messages that carry the information.

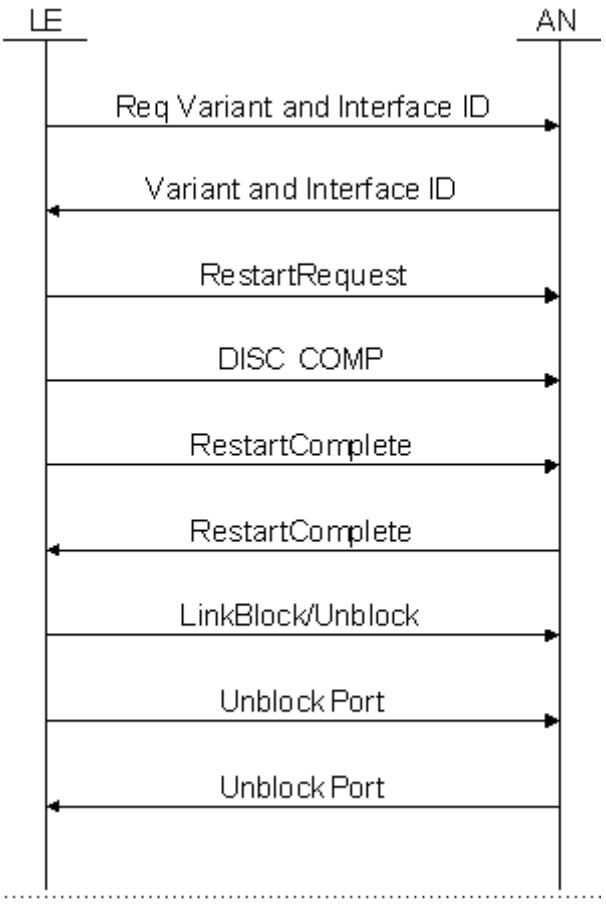
- The Common Control message is used for the Req Variant and Interface ID and the restart messages.
- The Disconnect Complete message is used for all of the configured user ports.
- The Unblock Port is a Port Control message.
- All of the messages are individually acknowledged (not shown).
- A PPL Config. Byte enables the restart procedure (see the *PPL Developer's Guide*). This is defaulted for the LE.

- At the end of the restart procedure is an option for choosing whether to send no link blocking and unblocking commands, to block only the links that were previously blocked, or to send the blocking and unblocking commands for each configured link.

4 Once the procedure is finished, the host receives a *PPL Event Indication* message indicating that startup is complete.

Important! Please see the *Appendix* for additional Call Flows pertaining to startup (Call Flows numbered 1-5).

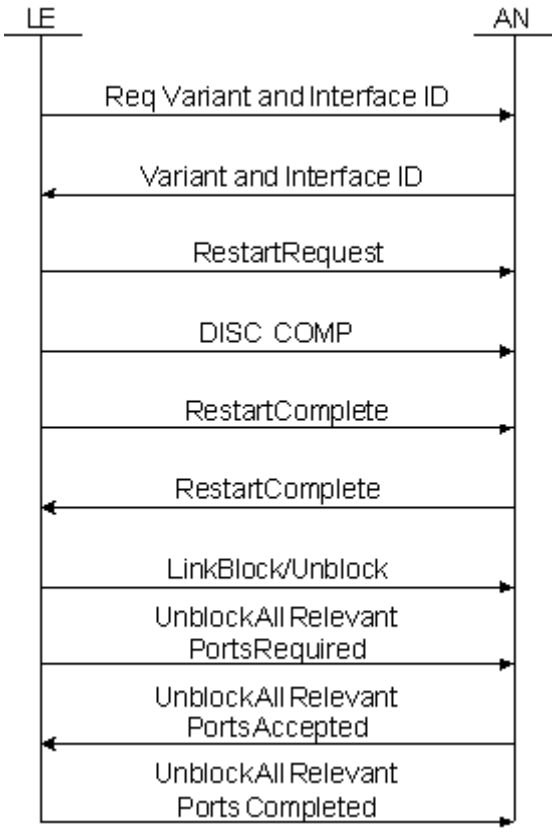
Typical Startup Sequence



Typical Startup Sequence with Accelerated Port Alignment

In the Typical Startup Sequence diagram above, it takes longer for all of the messages to propagate back and forth during this procedure.

- To speed startup, the Accelerated Port Alignment procedure is shown below.
- Instead of unblocking all ports individually, a new set of Common Control messages are sent that unblock all relevant ports.
- Then, either side can selectively block the user ports that it needs to block.



V5.2 Call Processing

Overview

Using the API messages only, the CSP supports V5 interface management protocols, and facilitates flexible call processing.

At configuration time, by using the Subscriber ID ICB, which uniquely identifies each subscriber, the host sets up Subscriber IDs by pairing a user port with a V5 ID.

To uniquely identify the source of a call, the CSP checks the status of the V5 Subscriber ID ICB for idle, busy, and blocked states. In the incoming call report, the *Request for Service with Data* message, the AIB is Expanded Span, Channel (0x0D). The V5 Subscriber ID ICB tells the host who the caller is, and the new V5 Formatted Data ICB carries the Information Elements (IEs) for the L3P PSTN protocol.

The *Route Control* API message is used for outgoing calls. The host must send the V5 Subscriber ID ICB for the destination subscriber. The physical timeslot is allocated dynamically, either through Layer 4 (for the LE) or from the LE (if the switch is acting as the AN). Only the LE selects the physical timeslot for V5.2. The *Route Control* response contains the selected logical span, channel. The host never selects the logical span, channel using V5.2. Whether it's the destination or Layer 4, the host will be told about only the channels that are used for the V5 side of the call. Span, channel selection occur conventionally, unless you use Dialogic's Call Control. Consistent with current applications, all call processing messages, including *Park Channel*, *Connect*, and *Connect to Conference*, use the Logical Span, Channel AIB (0x0D).

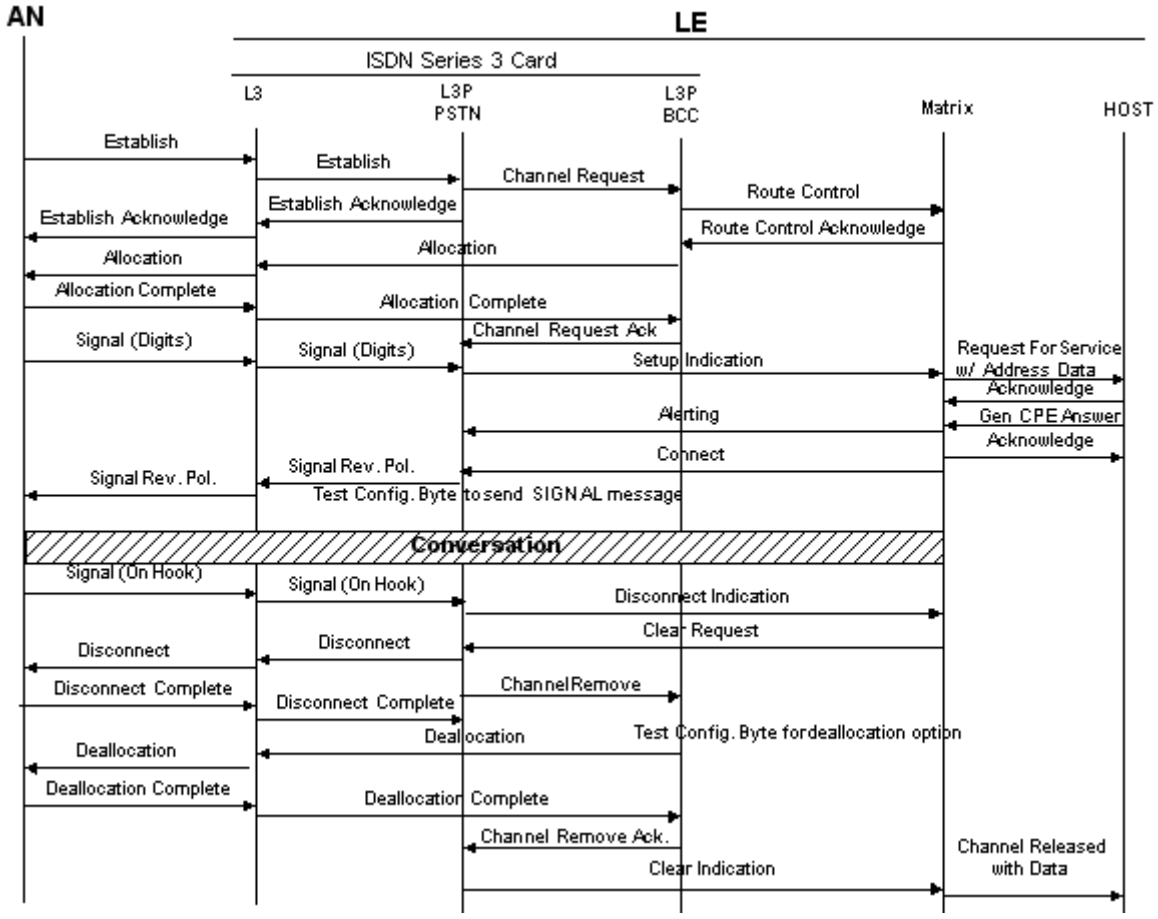
Important! This release of V5.2 supports LE only.

Important! AN is not supported for V5.2. Any text or illustrations (for example, call flows) indicating the AN side is for clarification purposes only.

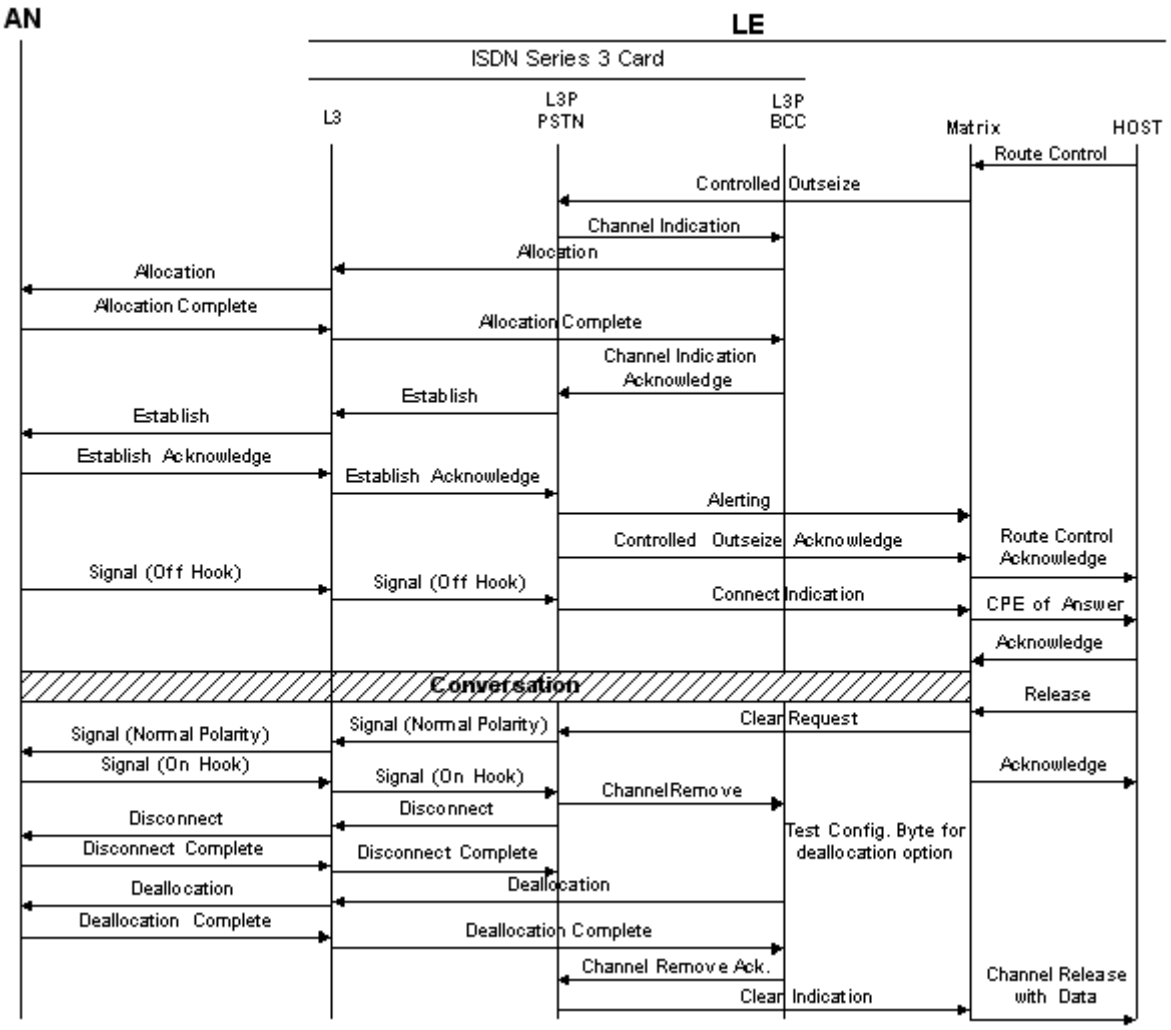
Sample Call Flows

This section presents diagrams that illustrate typical call flows.

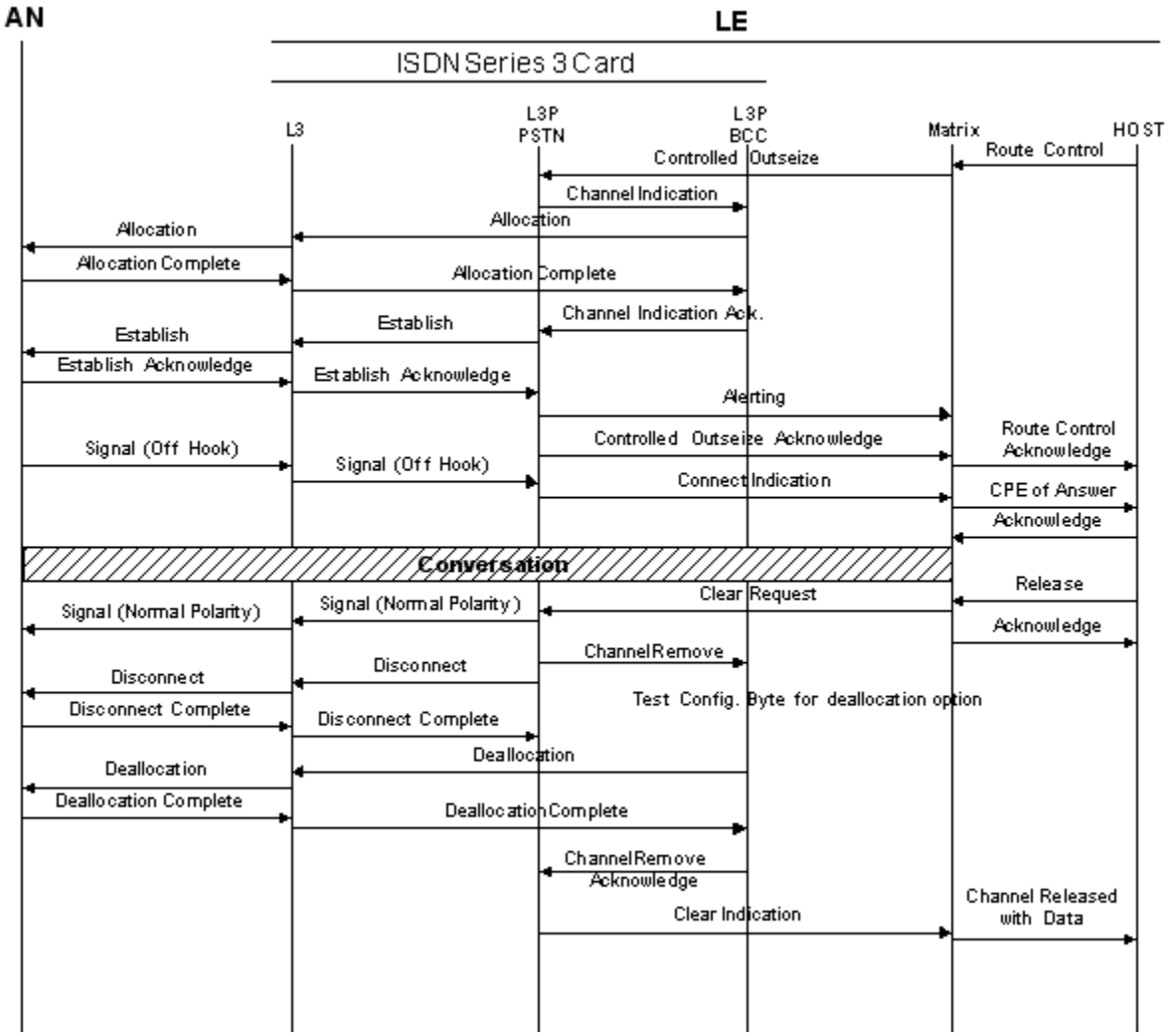
LE Side: Normal incoming call with subscriber release



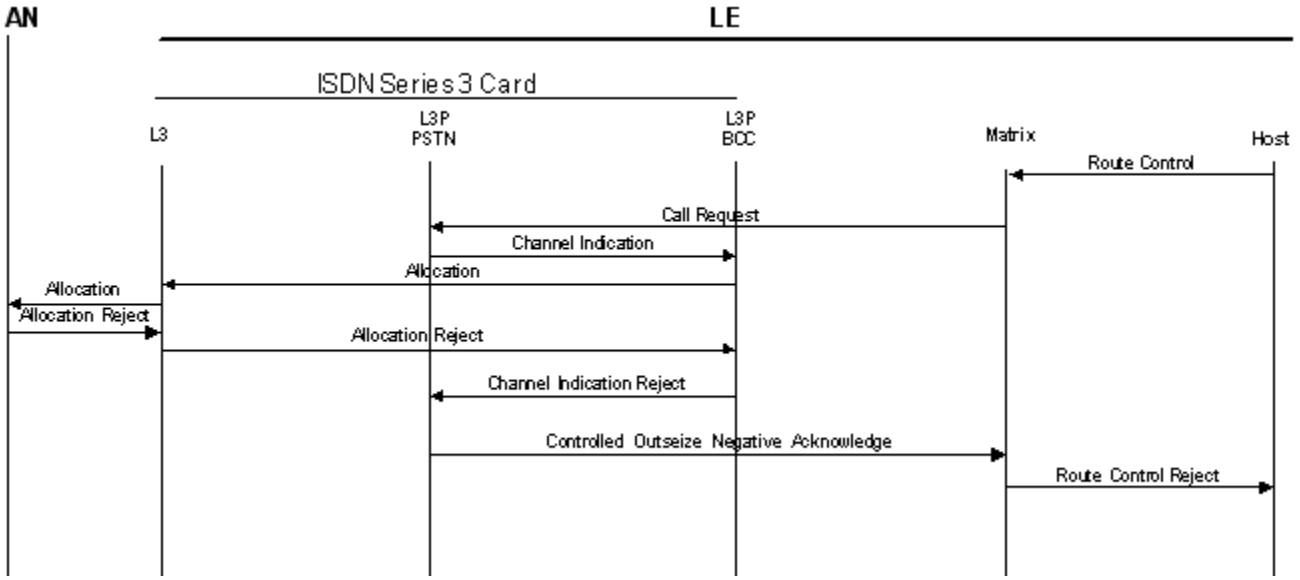
LE side: Outgoing call to AN with LE initiating on-hook



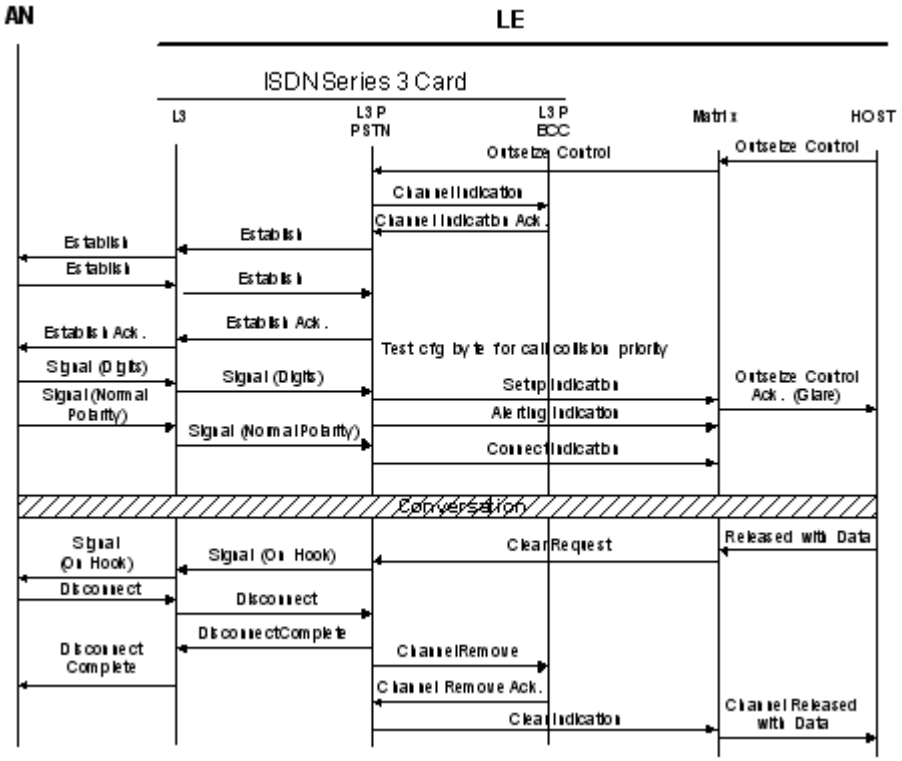
LE Side: Outgoing call to subscriber with LE initiating on-hook, but subscriber never hangs up



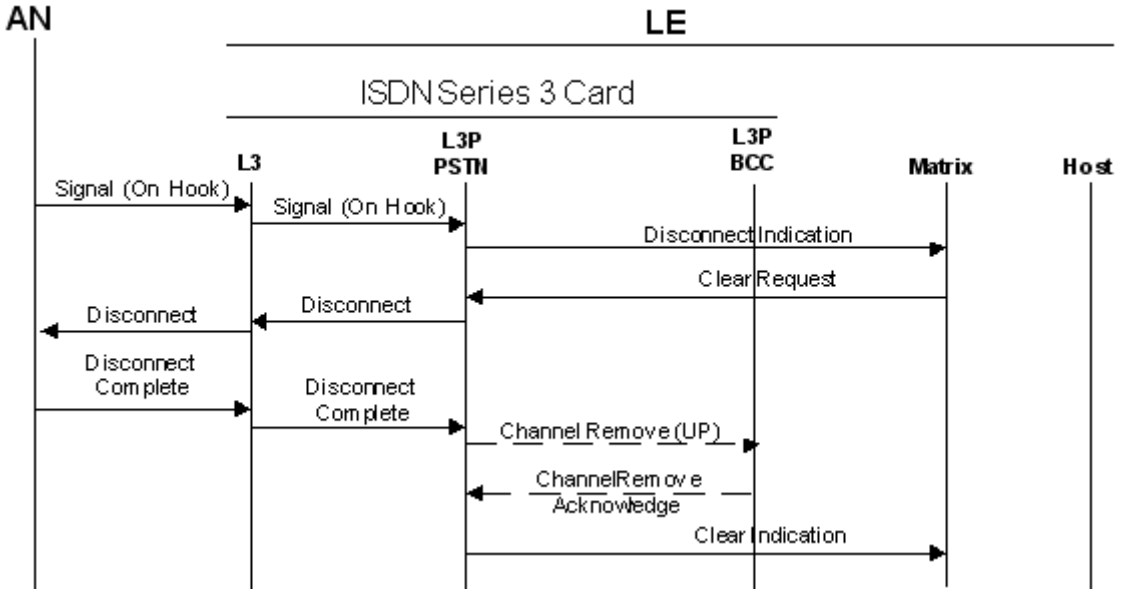
LE Side: Outgoing call with failed allocation



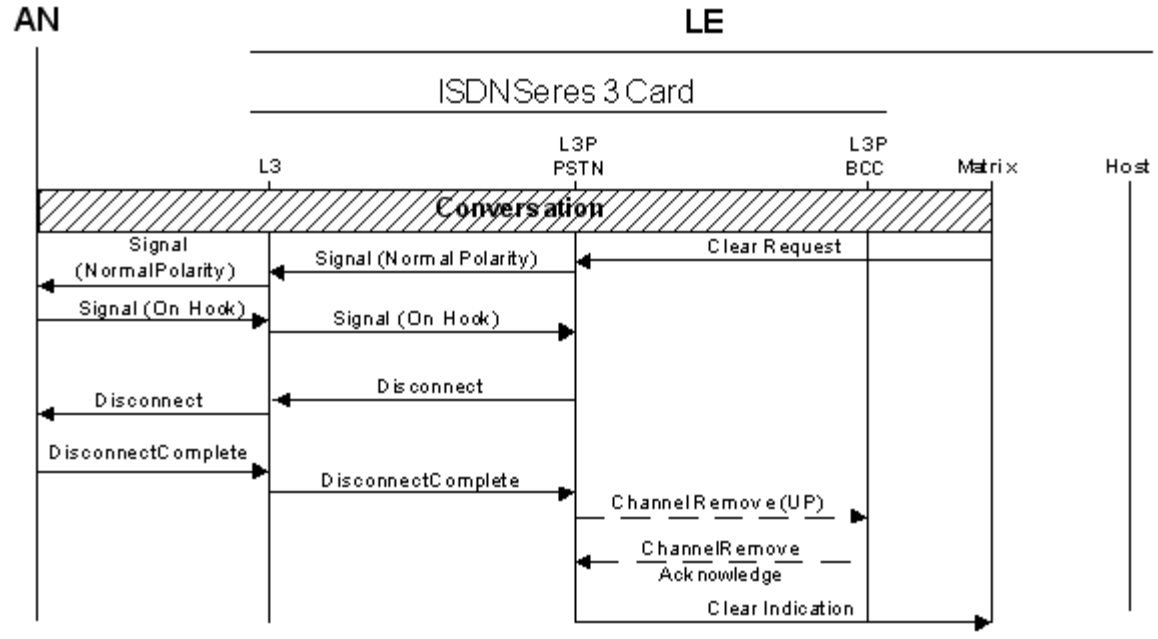
LE Side: Glare with incoming call having priority



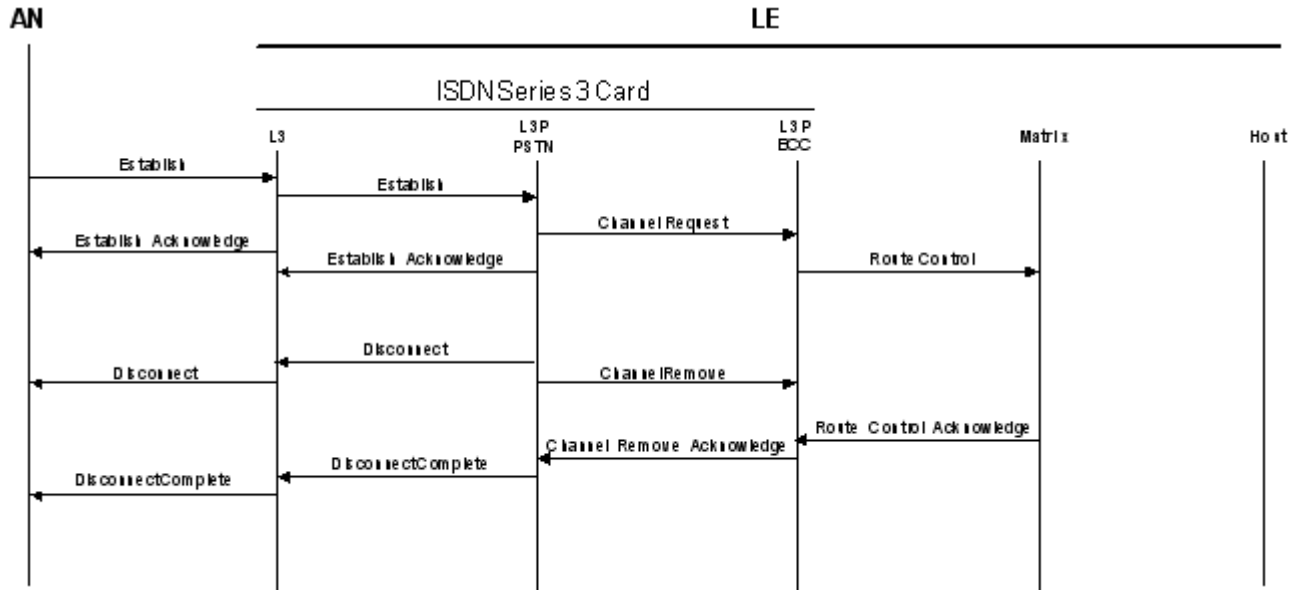
LE Side: Normal call clearing initiated by subscriber with timeslot deallocation options



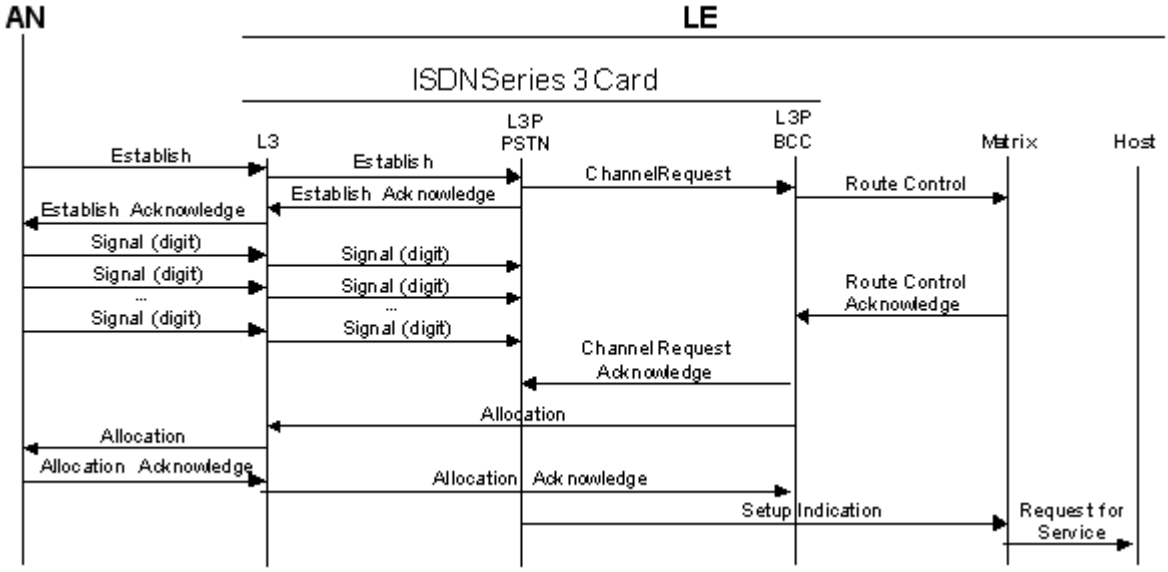
LE Side: Normal call clearing initiated by the LE side with timeslot deallocation options



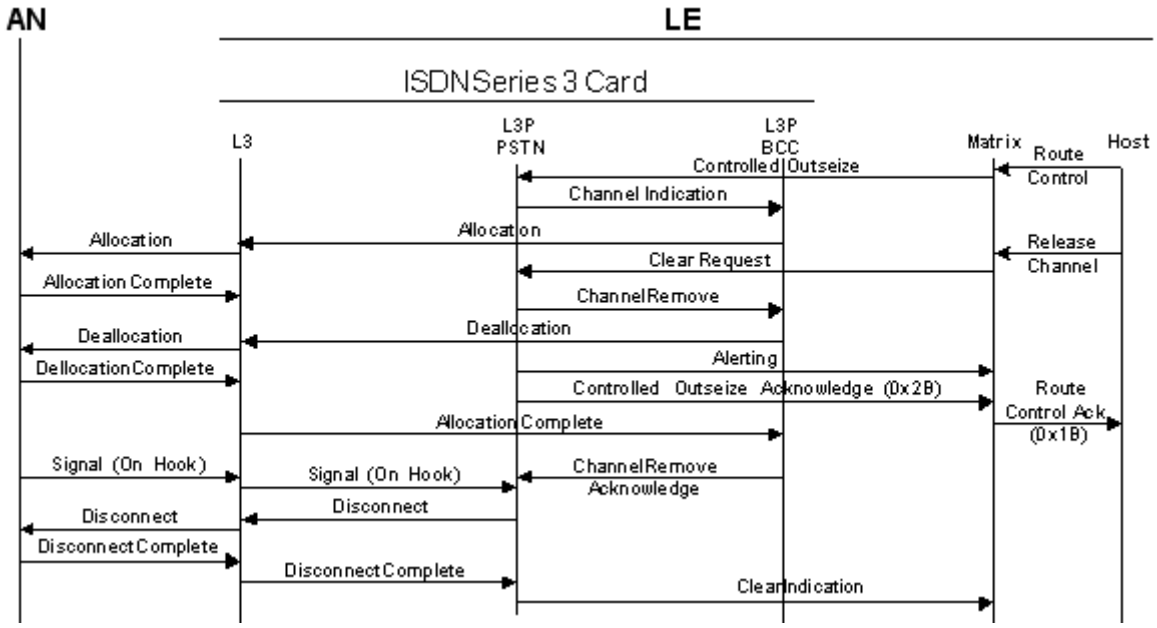
LE Side: Incoming call. AN releases before channel allocation



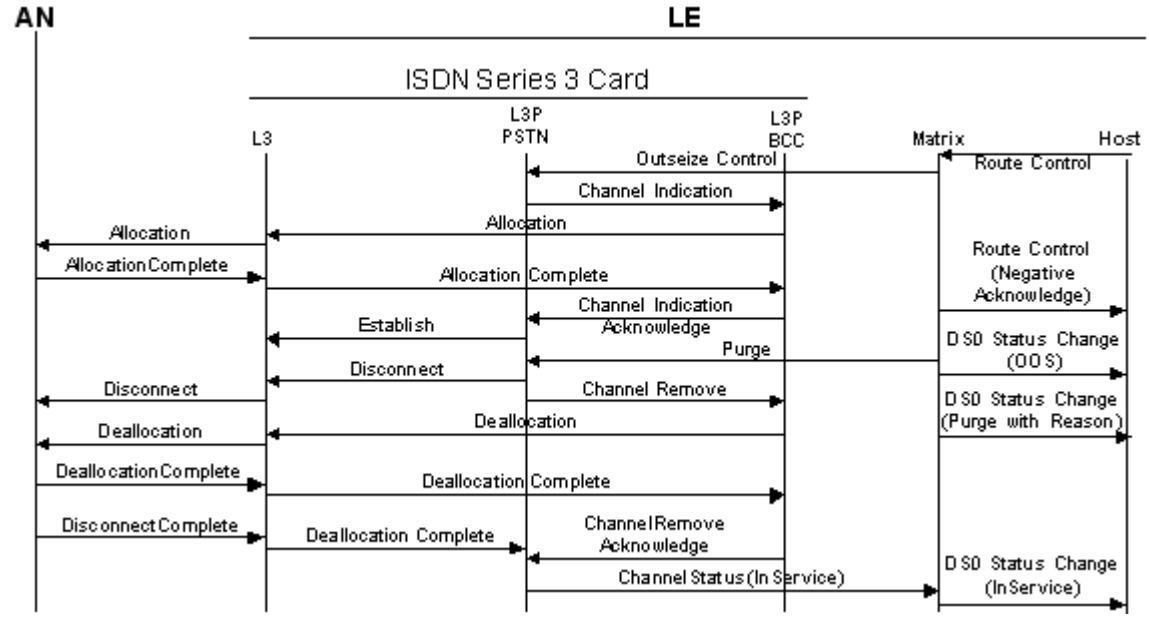
LE Side: Incoming call. Digits received before channel allocation



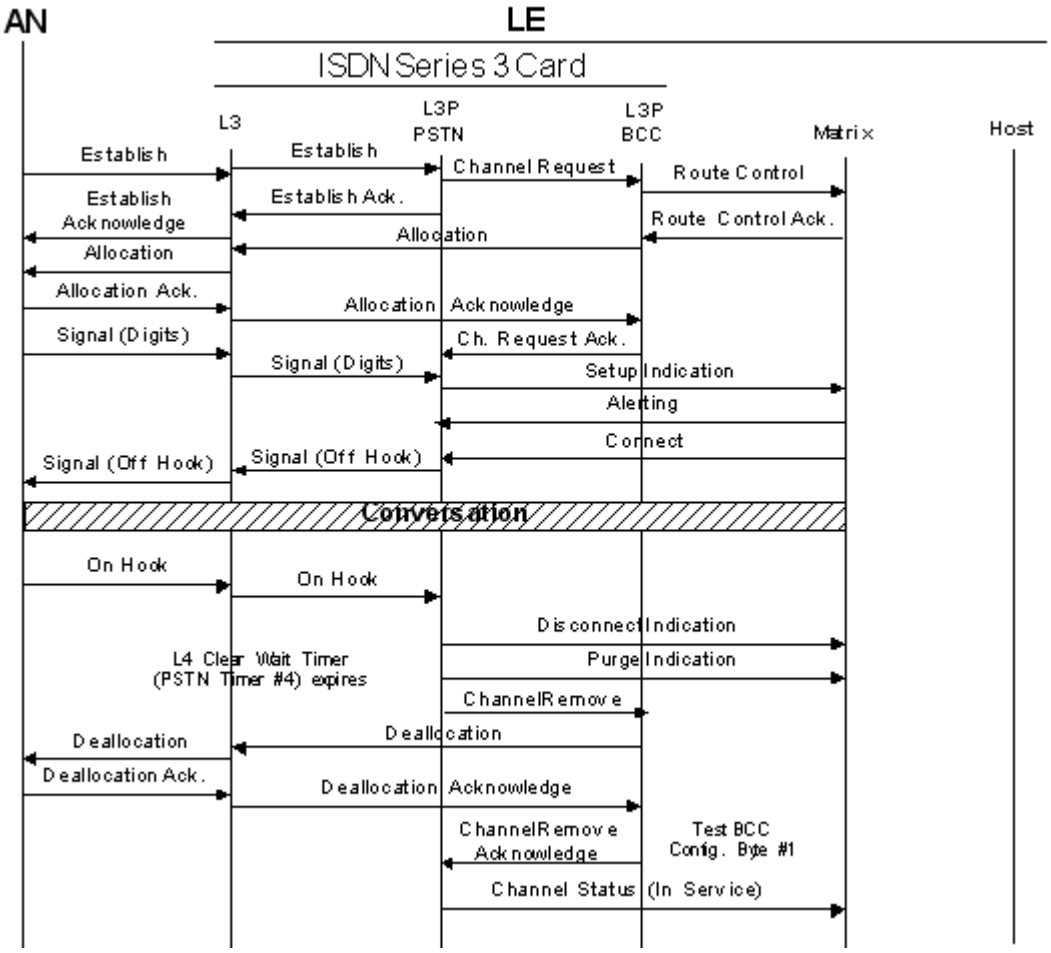
LE Side: Outgoing call. Cleared locally before allocation complete



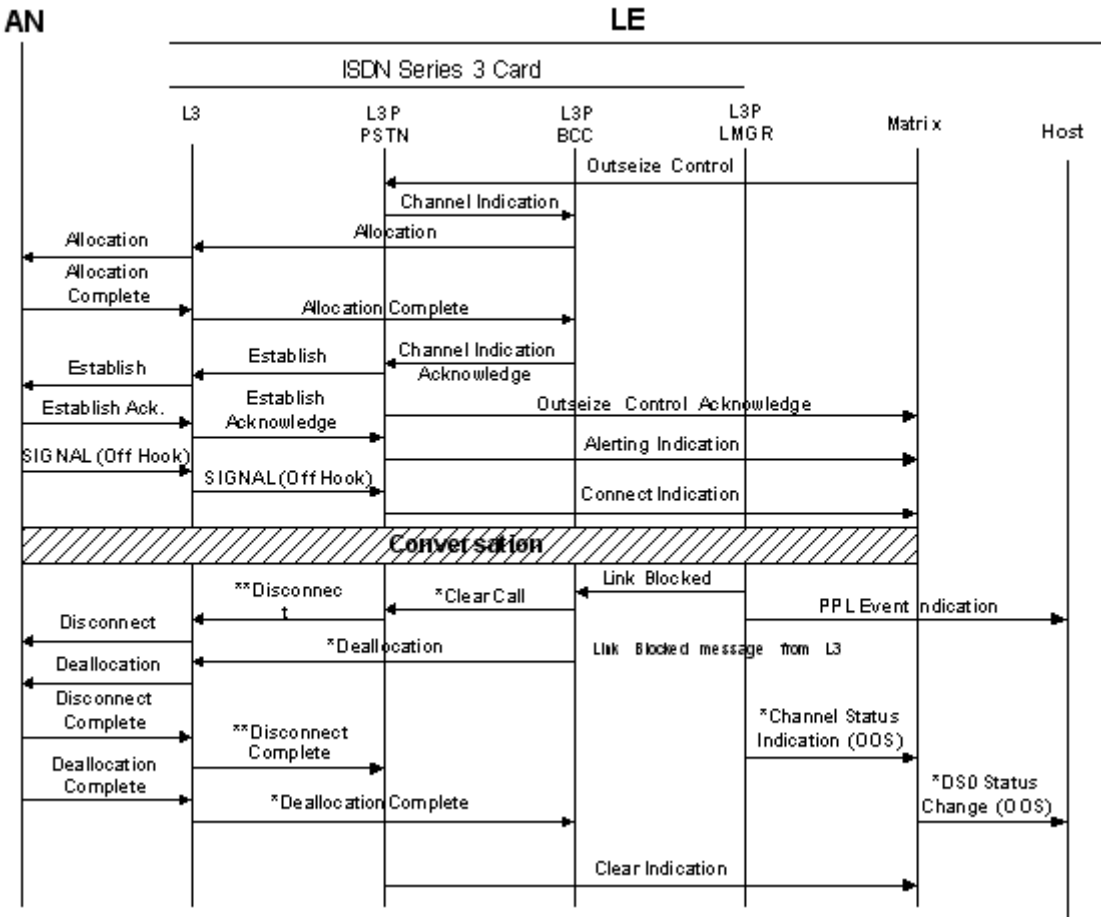
LE Side: Outgoing call with Layer 4-initiated PURGE



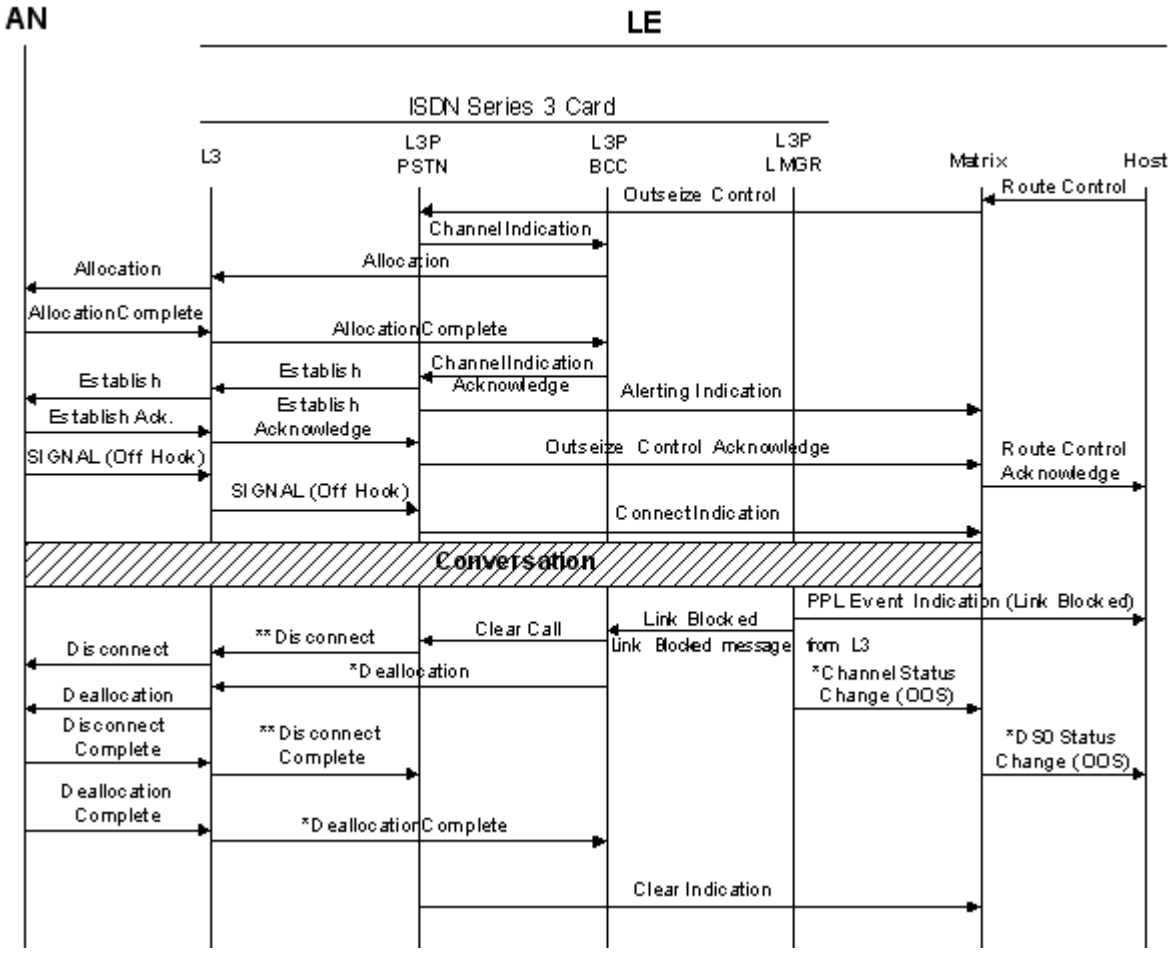
LE Side: Incoming call with Layer 3-initiated PURGE



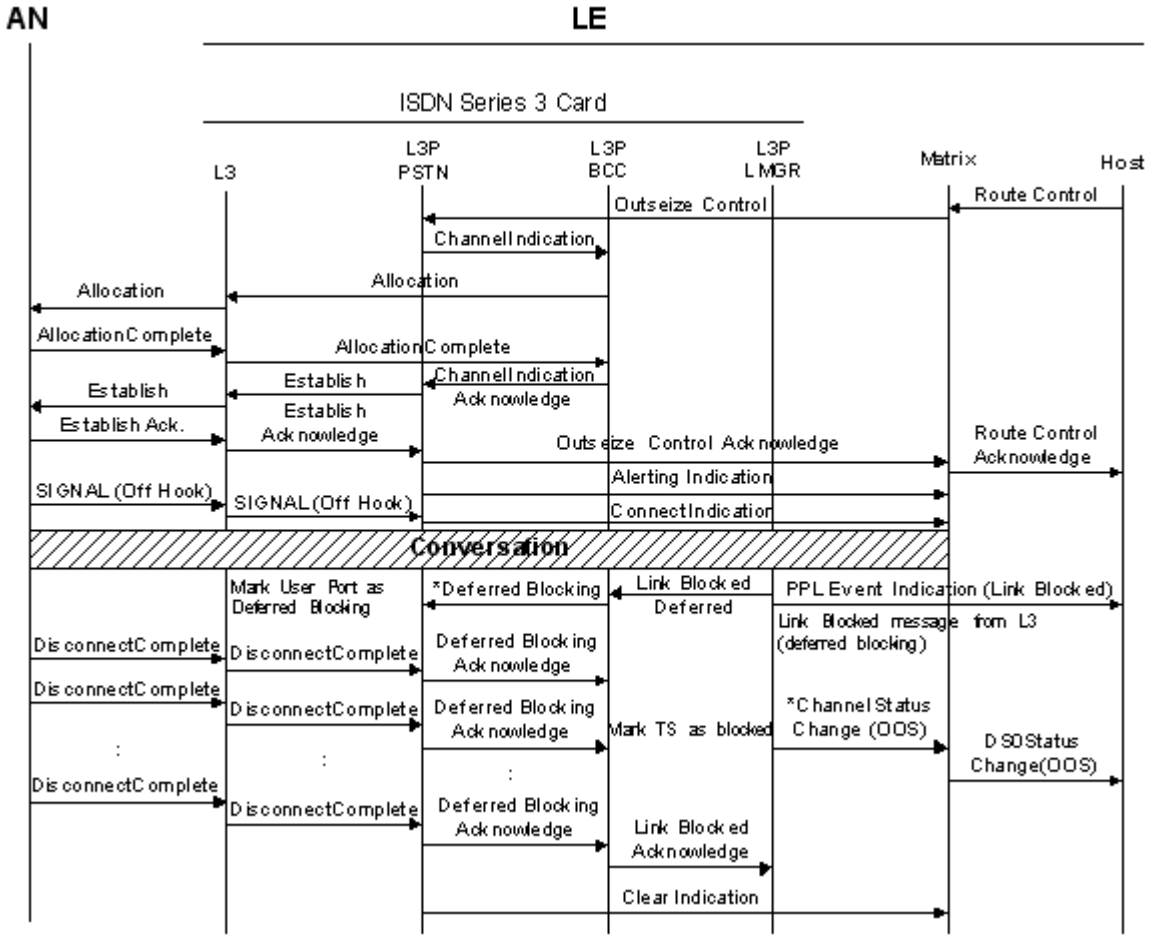
LE Side: Outgoing call with received Link Dead
*** Message sent for all channels on link**
****Message sent for all active calls on link**



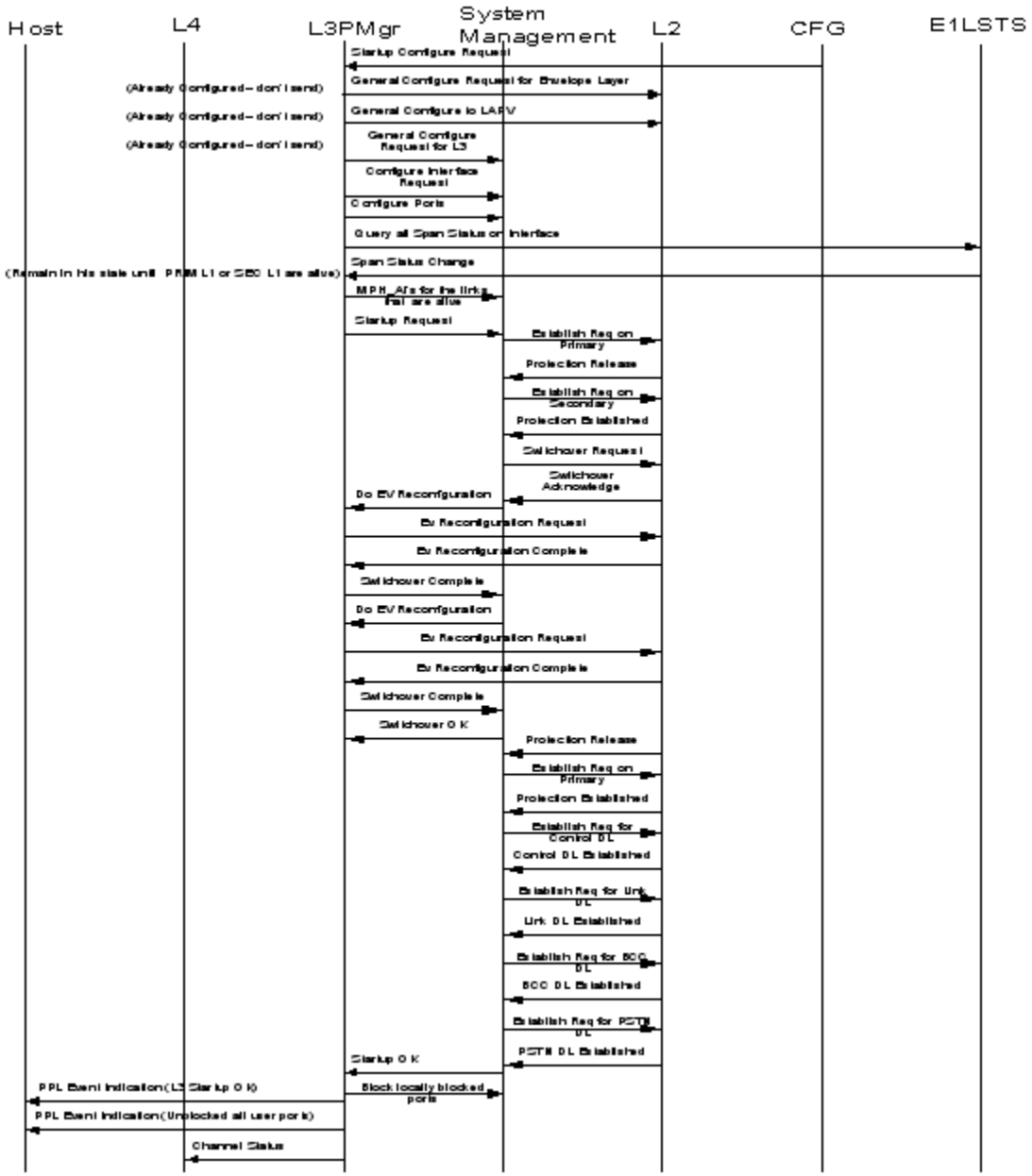
LE Side: Outgoing call with a received Link Blocking (Non-deferred)
*** These messages are sent for all channels on link**
****These messages are sent for all active calls on link**



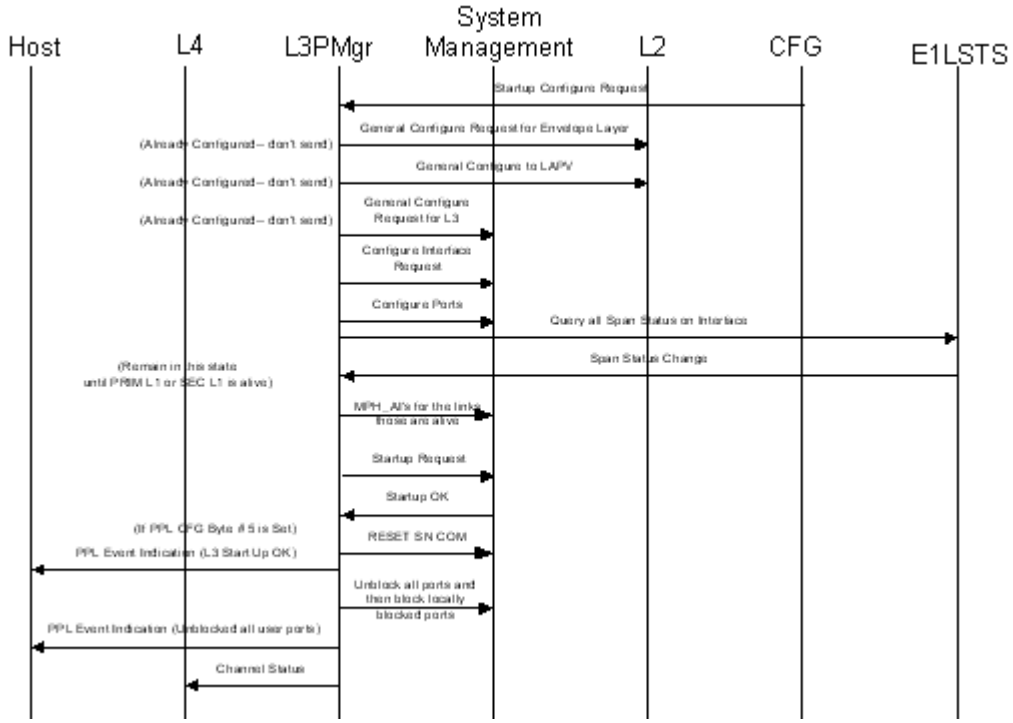
LE Side: Outgoing call with a received Link Blocking (Deferred)
***Messages sent on all channels on link**



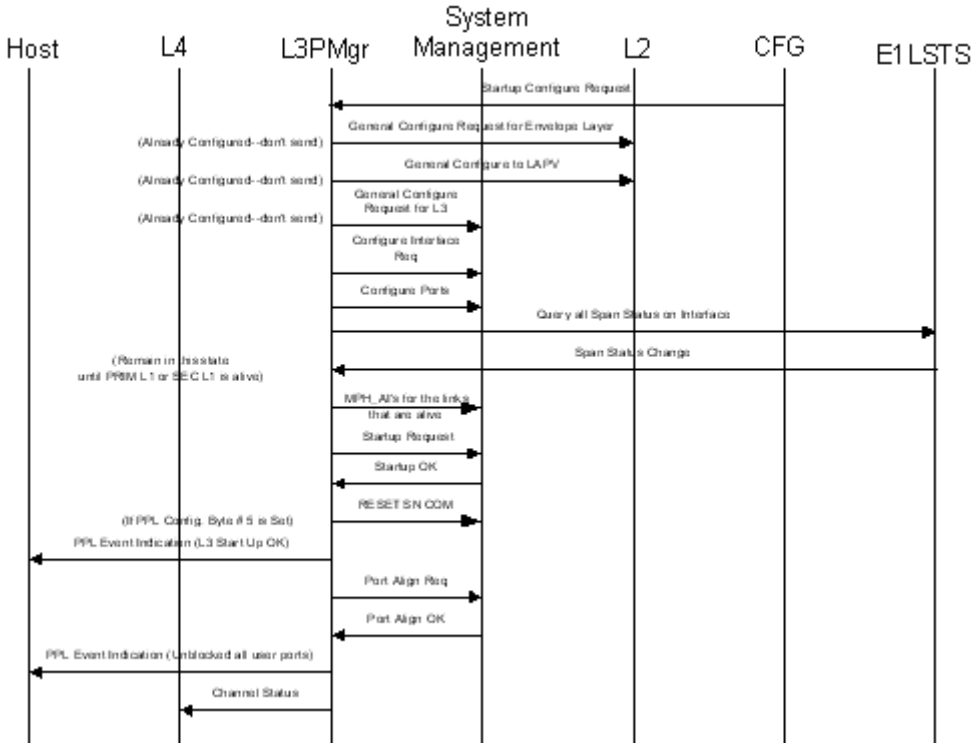
Startup with secondary link alive and primary still not up or dead



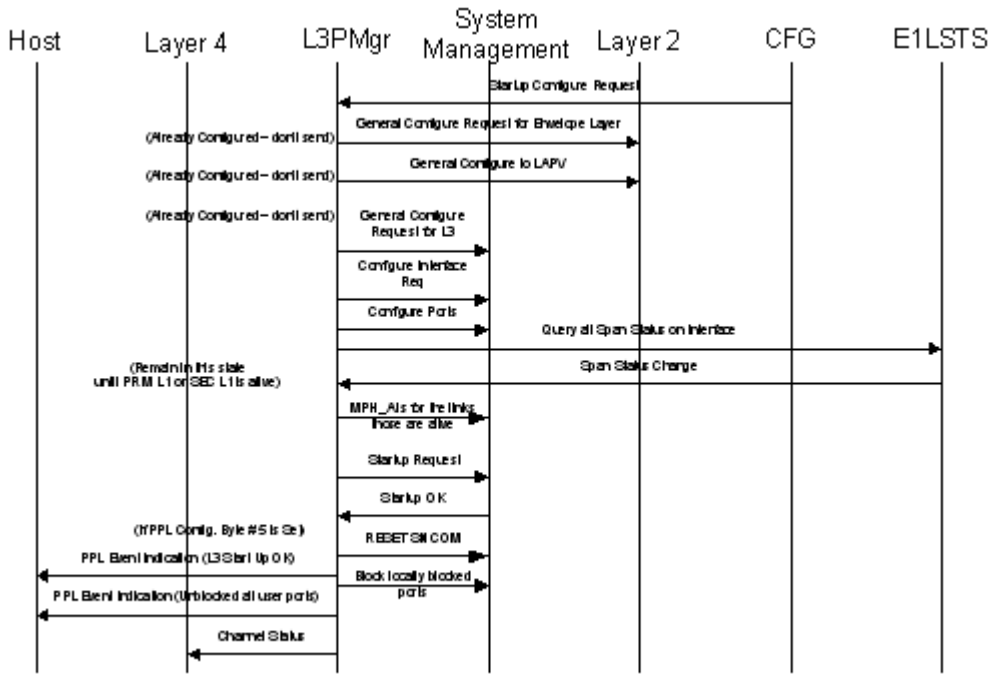
Startup with PPL Config. Byte #4 set to 0 (zero)



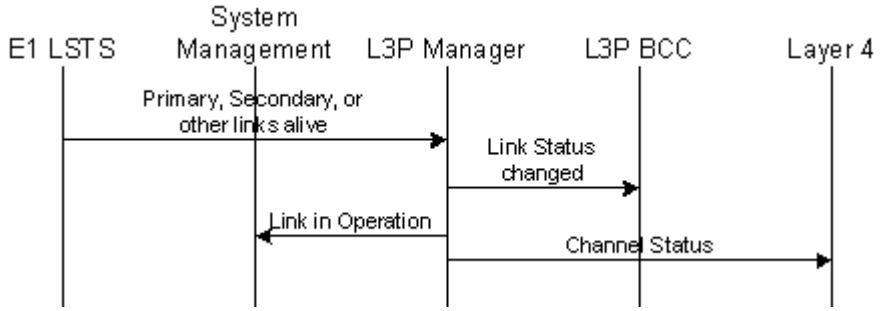
Startup with PPL Config. Byte #4 set to 1



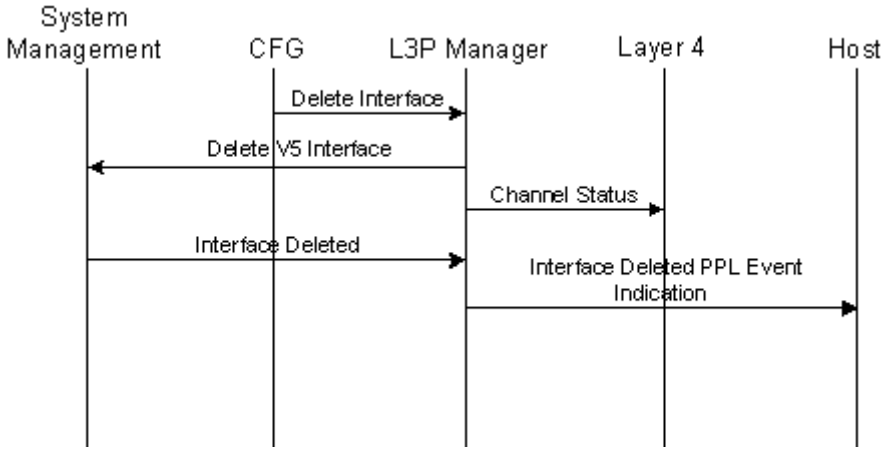
Startup with PPL Config. Byte #4 set to 2



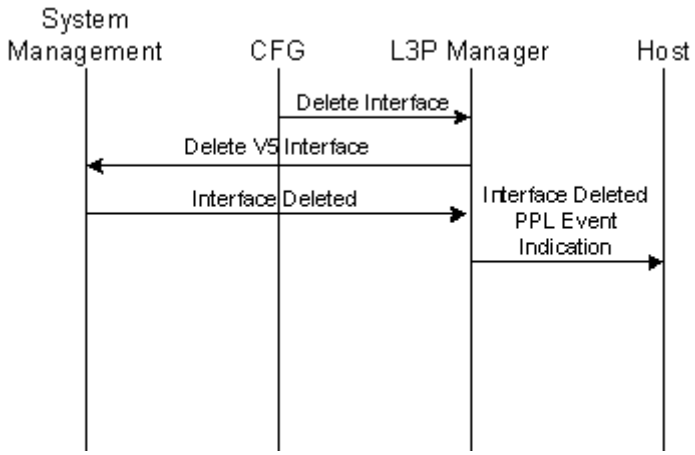
Received Link Alive while in Alive State



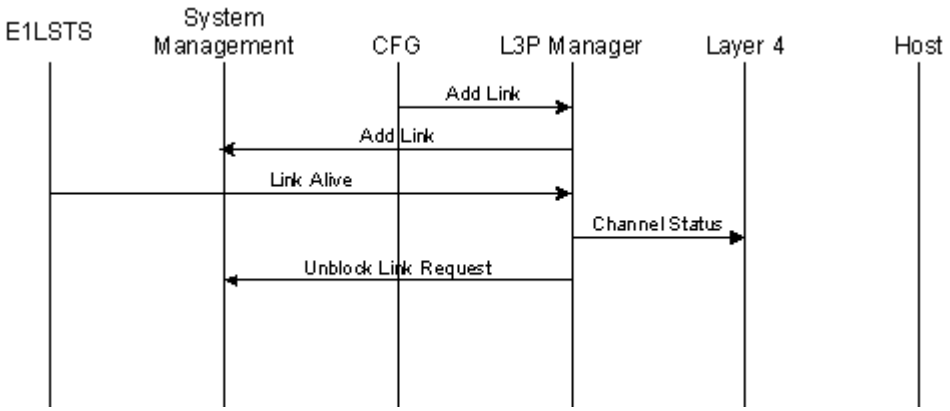
Delete Interface



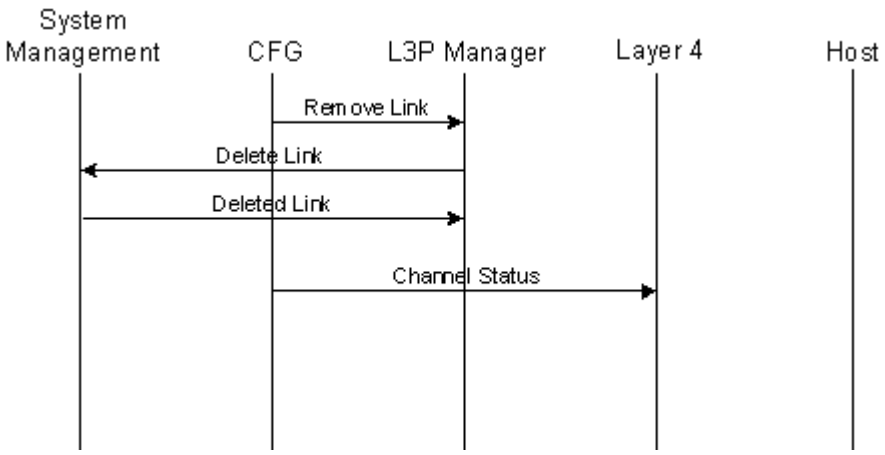
Delete Interface Forced



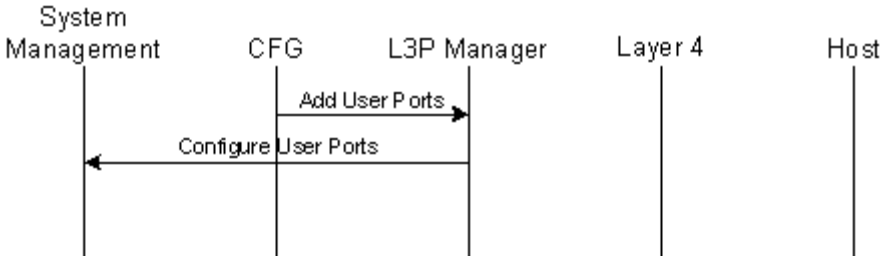
Received Add Link message from CFG



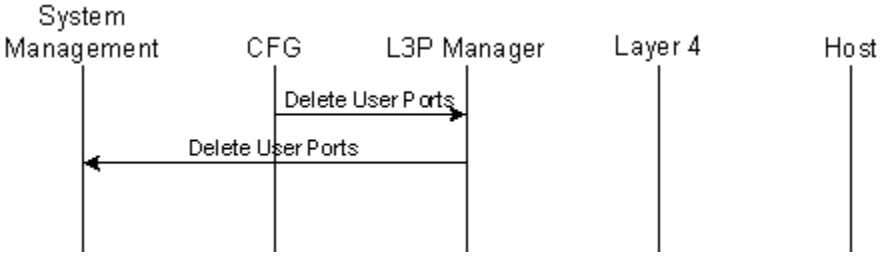
Remove Link



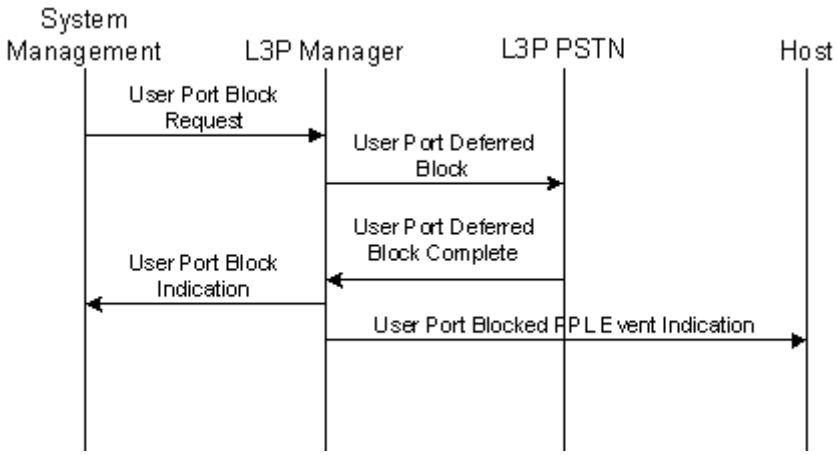
Add User Ports



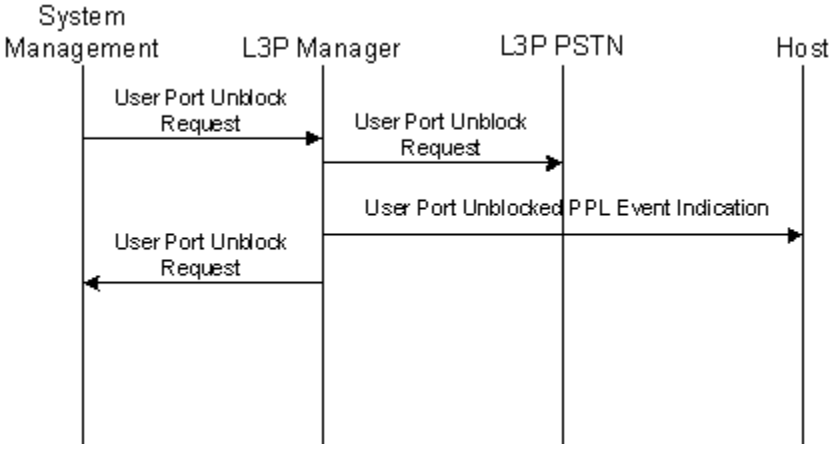
Remove User Ports



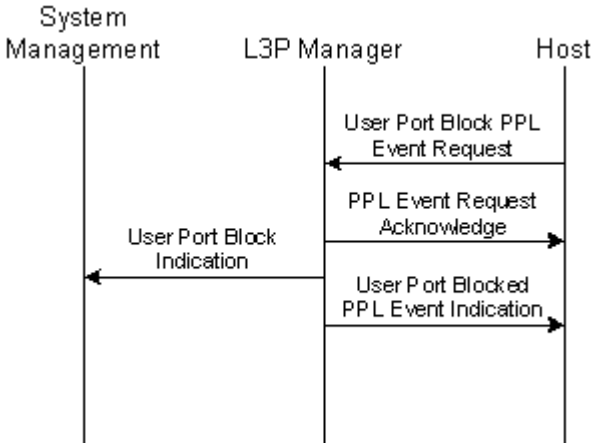
Received User Port Block request from Layer 3



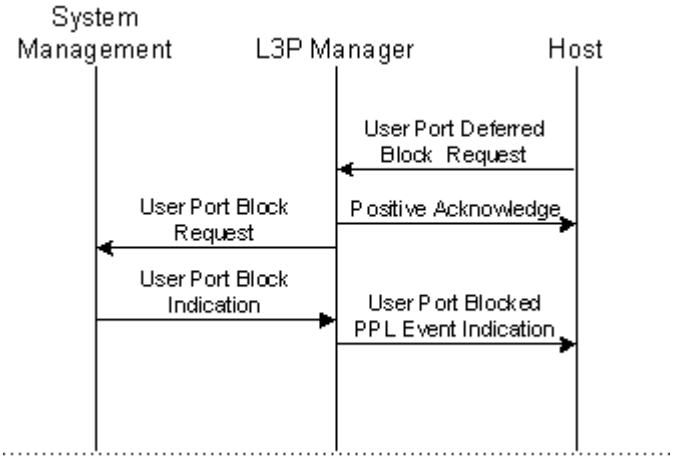
Received User Port Unblock request from Layer 3



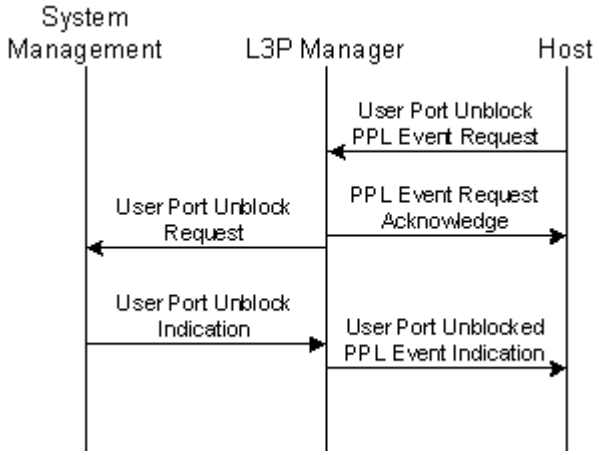
Received User Port Block request from host



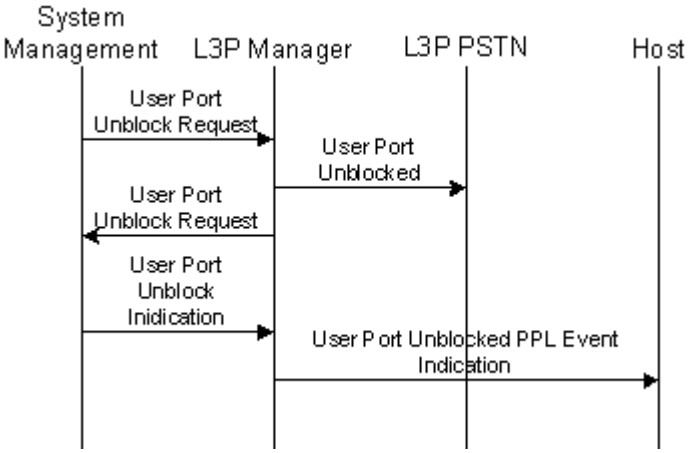
User Port Deferred Block request



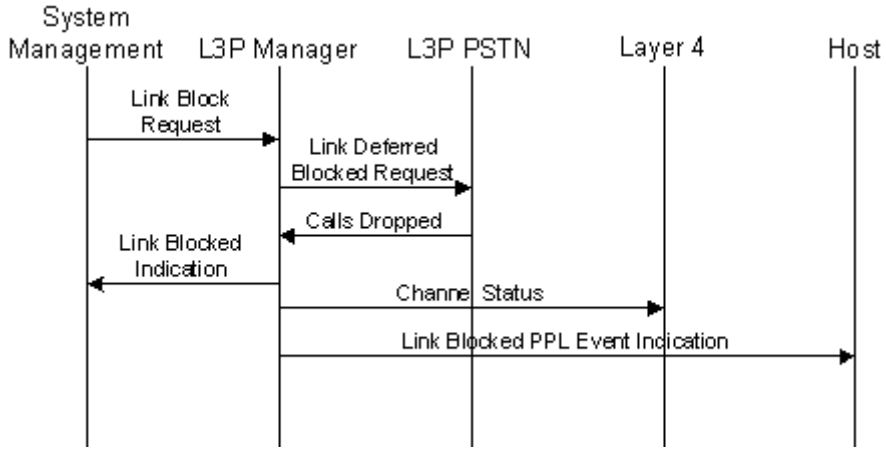
Received user Port Unblock request from host



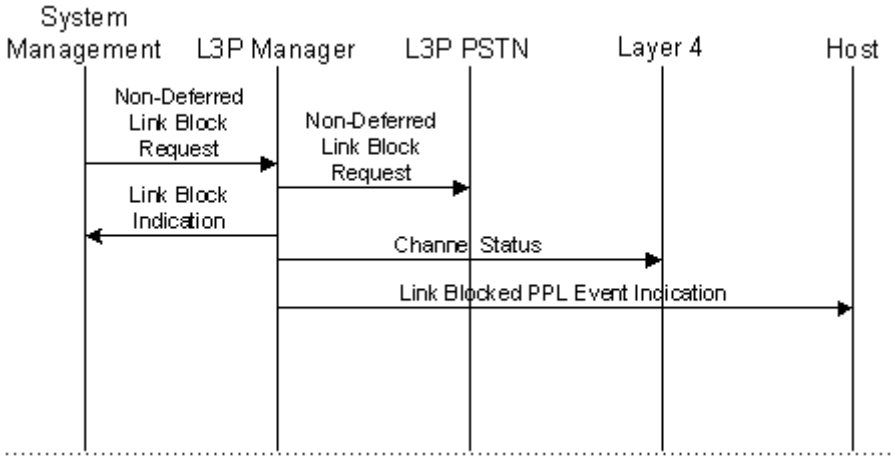
User Port Unblock request from System Management



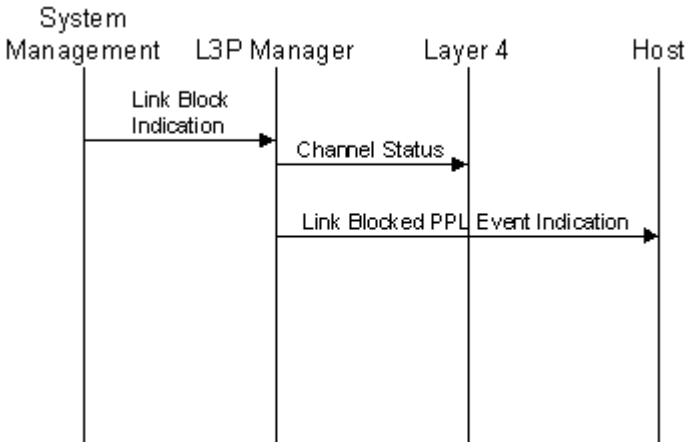
Link Block request from System Management



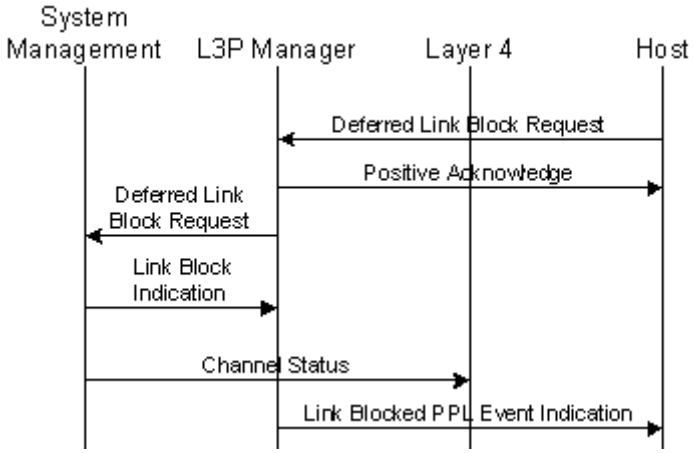
Non-Deferred Link Block request



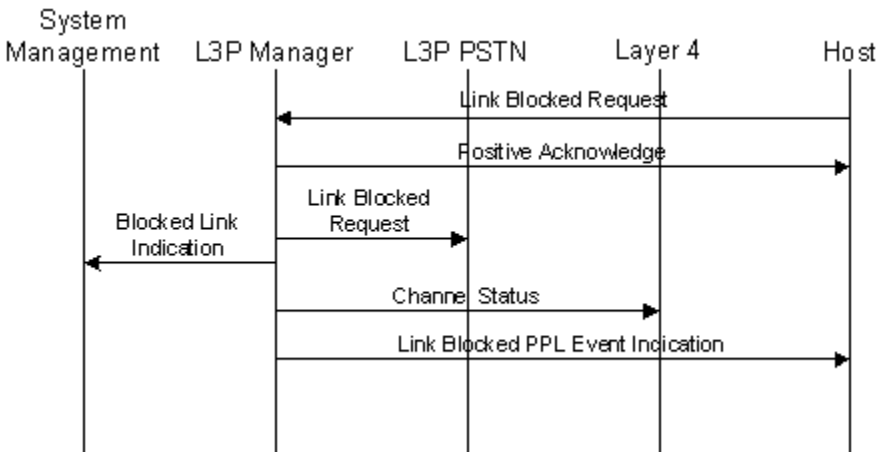
Link Block Indication



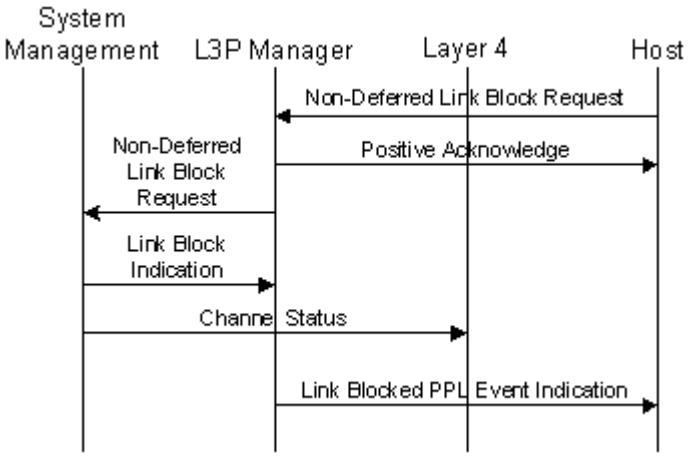
Deferred Link Block request



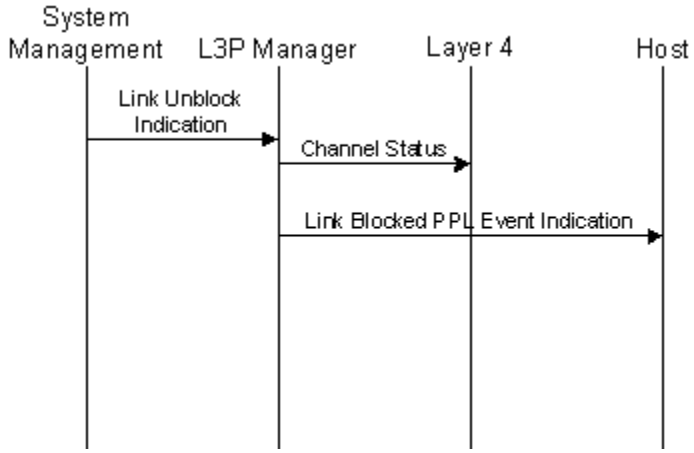
Received Link Blocked request from host



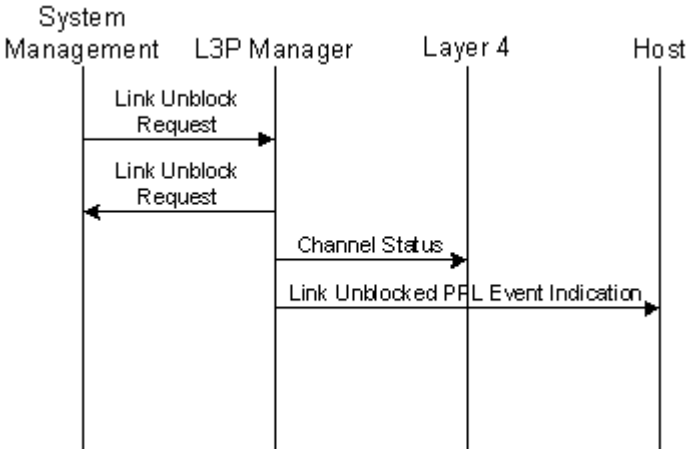
Non-Deferred Link Block request from host



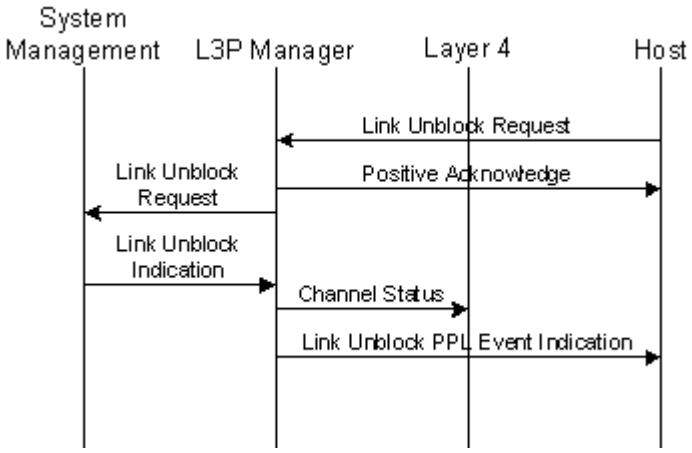
Link Unblock Indication from System Management



Link Unblock request from System Management

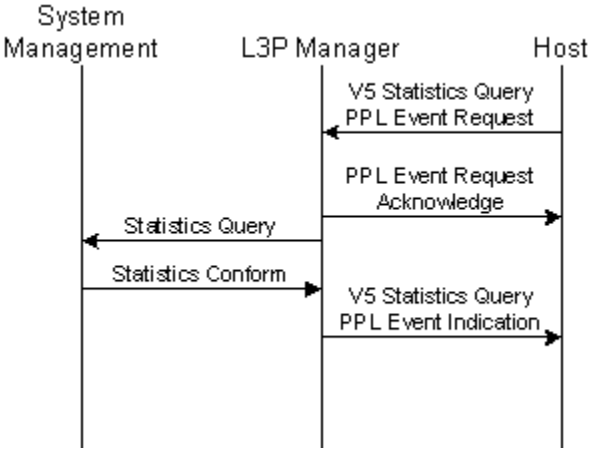


Link Unblock request from host



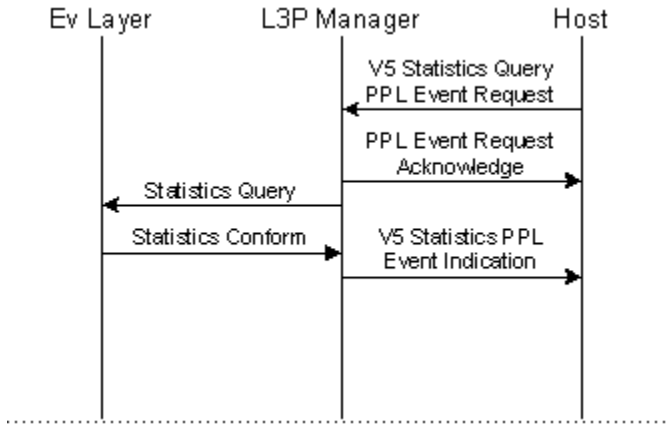
V5 Statistics Query from host, Case 1:

Statistics Query for PSTN, BCC, CONTROL, PROTECTION, LINK Protocols and System Management



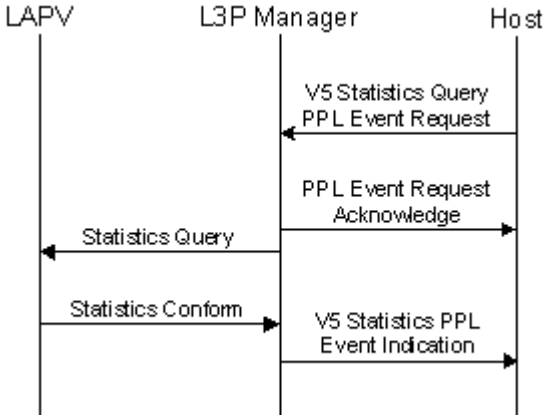
V5 Statistics Query from host, Case 2:

Statistics Query for Envelope Layer

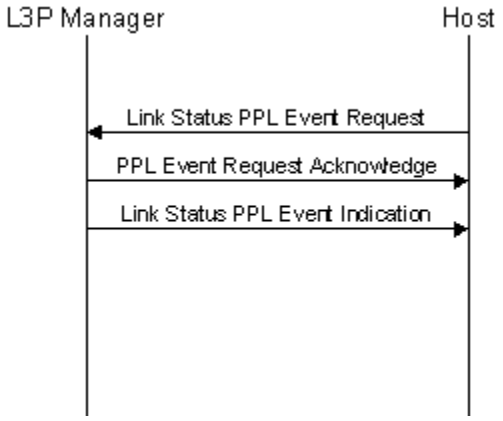


V5 Statistics Query, Case 3:

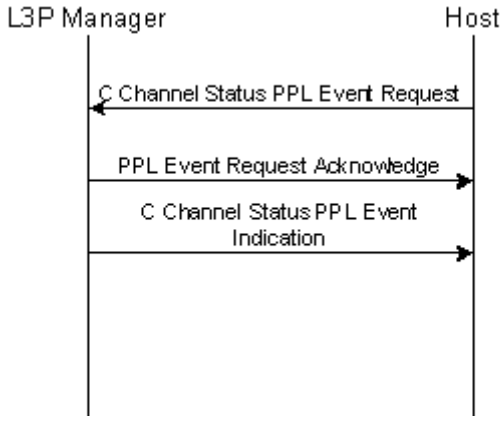
Statistics Query for LAPV



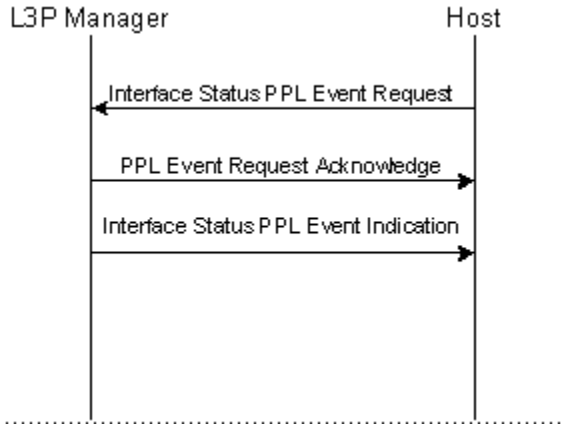
Received Link Status Query



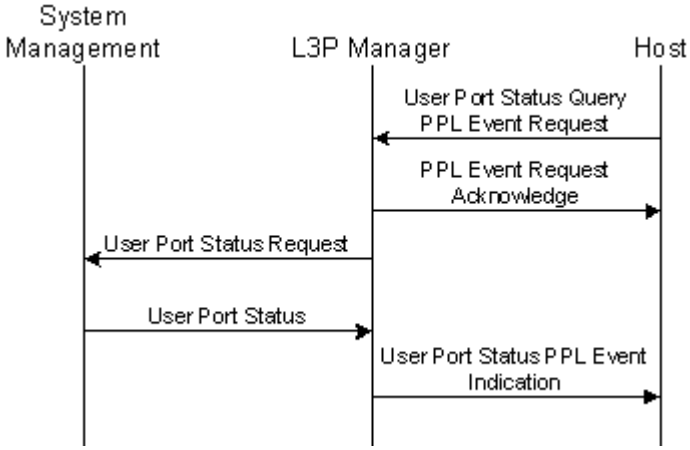
C Channel Status query



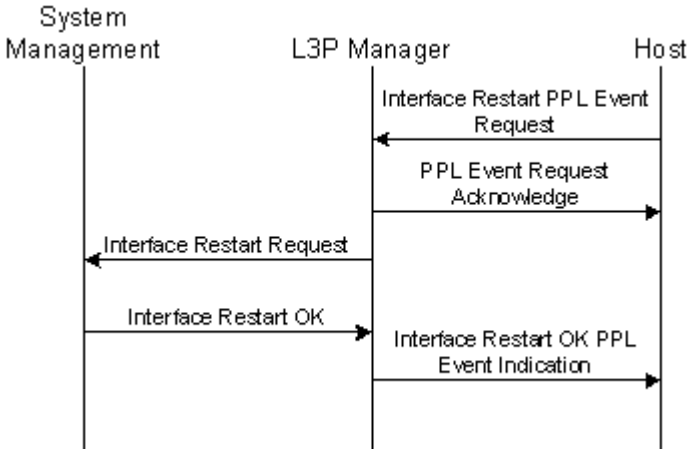
V5 Interface Status query



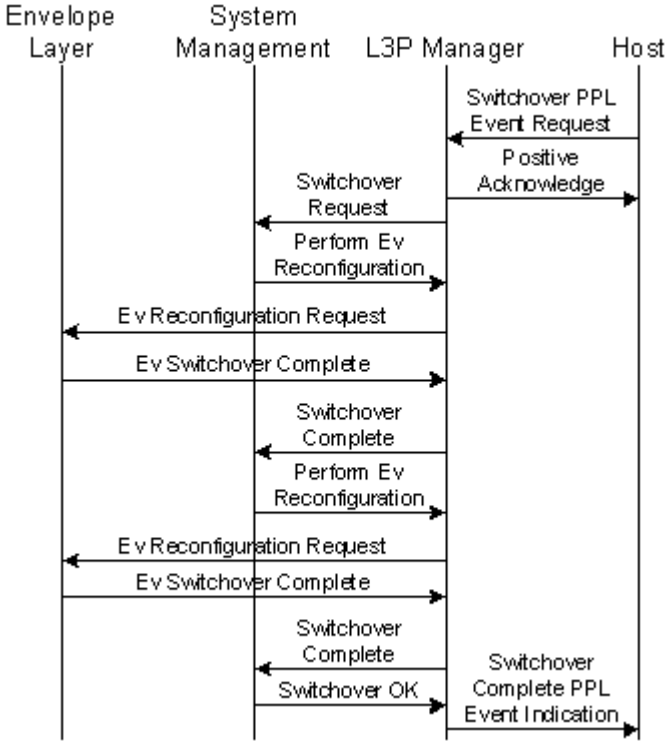
User Port Status request



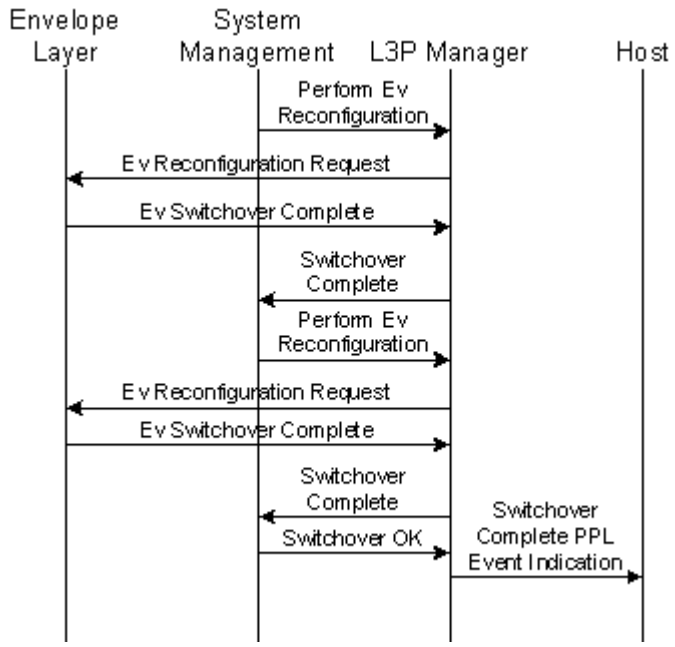
V5 Interface Restart request



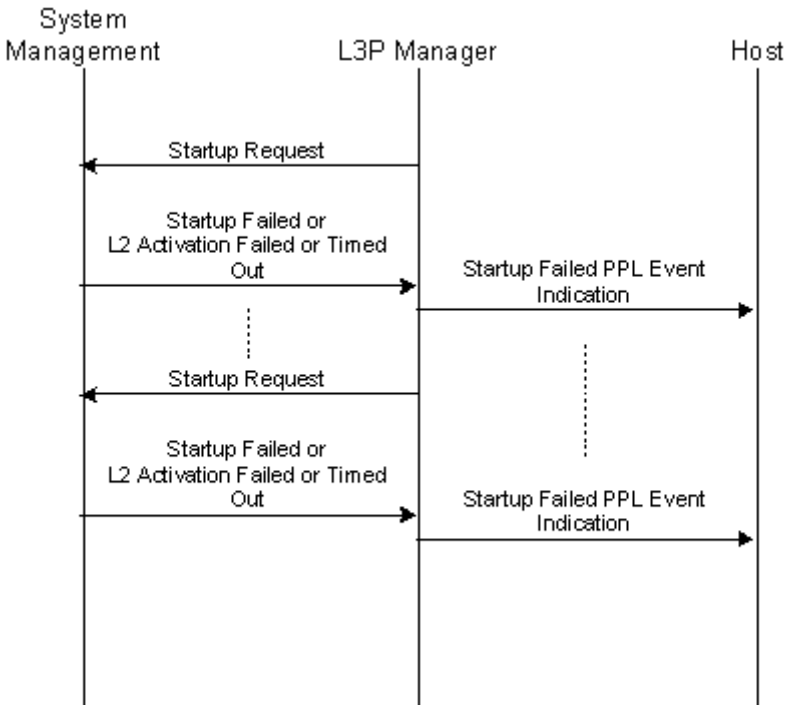
Switchover request from host



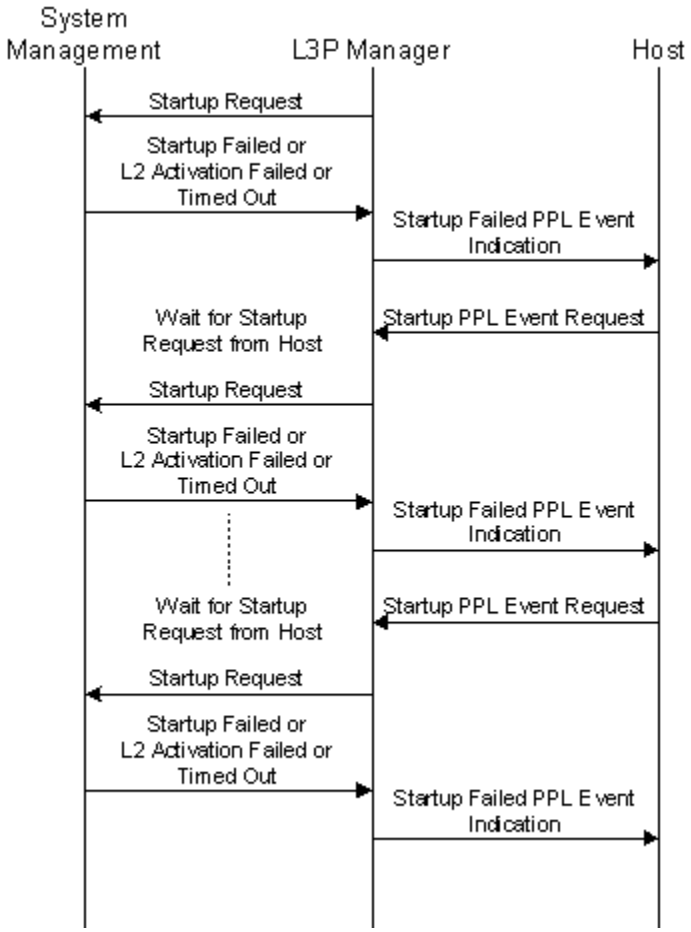
Switchover request from System Management



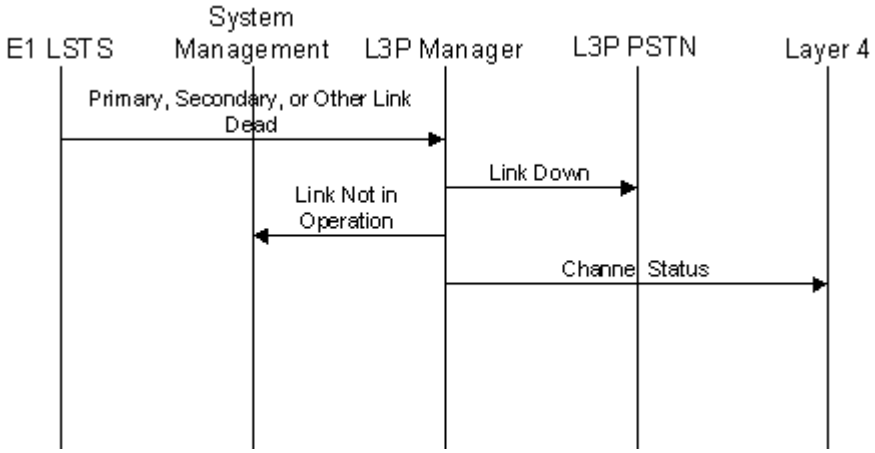
Startup failed and PPL Config. Byte 3 set to 0 (zero)



Startup failed and PPL Config. Byte 3 set to 1 (one)

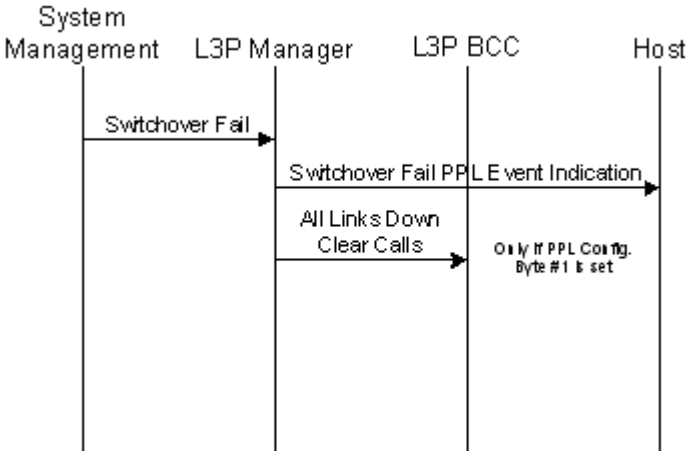


Link Dead



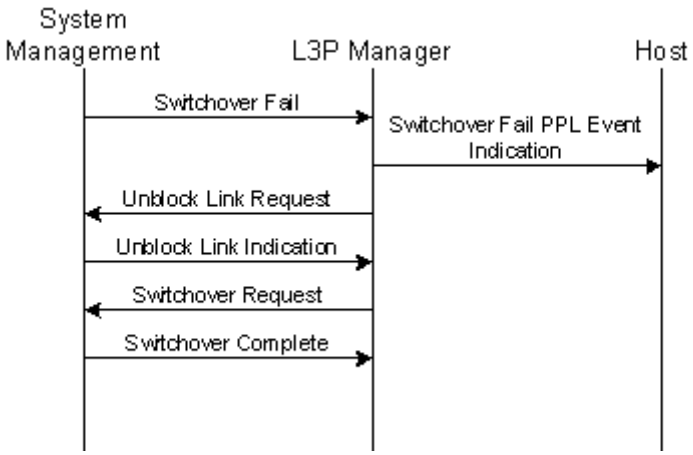
Switchover fail received from System Management, Case 1:

Both C Channels dead



Switchover fail received from System Management, Case 2:

One C Channel is blocked, and the other C Channel is dead



13 V5.2 PPL Information

Purpose This chapter provides PPL information for V5.2.

For a list of the PPL Components and addressing information, see *PPL Component IDs* and *PPL Component Addressing* in the *API Reference*.

Important! This release of V5.2 supports LE only.

L3P PSTN Component (0x8B)

Purpose

This section describes the PPL timers for L3P PSTN Component.

L3P PSTN Component PPL Timers (0x008B)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Establish Acknowledge Wait (for first Establish only)	2 seconds
0x02	Establish Acknowledge Wait (for further Establish resends)	5 seconds
0x03	Disconnect Complete Wait	2 seconds
0x04	Status Wait	2 seconds
0x05	Signal or Protocol Parameter Received	100 ms
0x06	Signal Acknowledge Wait	10 seconds
0x07	Waiting for Disconnect Response	4 seconds
0x08	Waiting for Disconnect Request	4 seconds
0x09	Waiting for Datalink Connection Establish	4 seconds

L3P BCC Protocol (0x8C)

Purpose

This section describes the PPL timers for L3P BCC Protocol.

L3P BCC Protocol PPL Timers (0x008C)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Allocation Complete Wait	1.5 seconds
0x02	Deallocation Complete Wait	2 seconds
0x03	Deallocation Complete Wait (in Null state)	2 seconds
0x04	Audit Complete Wait	1.5 seconds
0x05	AN Fault Acknowledge Wait	1.5 seconds
0x06	Waiting for Datalink Connection Establish	4 seconds

L3P BCC Link Protocol (0x8D)

Purpose

This section describes the PPL timers for L3P Link Protocol.

L3P Link Protocol PPL Timers (0x008D)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Link Control Message Wait	1 seconds
0x02	Waiting for Datalink Connection Establish	4 seconds

L3P Protection Protocol (0x8E)

Purpose

This section describes the PPL timers for L3P Protection Protocol.

L3P Protection Protocol PPL Timers (0x008E)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Waiting for Switchover Acknowledge	1.5 seconds
0x02	Waiting for Switchover Acknowledge	1.5 seconds
0x03	Waiting for Switchover Com	1.5 seconds
0x04	Waiting for Reset SN Acknowledge	20 seconds
0x05	Will always expire	10 seconds

L3P Control Protocol (0x8F)

Purpose

This section describes the PPL timers for L3P Control Protocol.

L3P Control Protocol PPL Timers (0x008F)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Waiting for Port Control Acknowledge	1 seconds
0x02	Waiting for Common Control Acknowledge	1 seconds
0x03	Waiting for Datalink Connection Establish	4 seconds

L3P System Management Component (0x90)

Purpose

This section describes the PPL timers for L3P System Management Component.

L3P System Management Component PPL Timers (0x0090)

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Waiting for MDU Control (restart acknowledge) from all PSTN protocol State Machines	100 seconds
0x02	Waiting for MDU Control (restart complete) from Control Data Link	120 seconds
0x03	Waiting for MDL Establish Confirm or MDL Establish Indication from Control Data Link	15 seconds
0x04	Waiting for MDL Establish Confirm or MDL Establish Indication from Control Data Link	60 seconds
0x05	Waiting for MDL Establish Confirm or MDL Establish Indication from PSTN Data Link	25 seconds
0x06	Waiting for MDL Establish Confirm or MDL Establish Indication from Link Control Data Link	15 seconds
0x07	Waiting for MDL Establish Confirm or MDL Establish Indication from Link Control Data Link	60 seconds

Timer ID	Description	Value (default)
0x08	Waiting for MDL Establish Confirm or MDL Establish Indication from BCC Data Link	15 seconds
0x09	Waiting for MDU Control (Unblock all relevant ports accepted)	20 seconds
0x0A	Waiting for MDU Control (Unblock all relevant ports accepted)	10 seconds

L3P PSTN (0x91)

Purpose L3P PSTN uses PPL configuration byte, event, and timer information.

L3P PSTN Configuration Bytes

Byte	Description	Values (* Default)
0x01	Send optional SIGNAL message (with steady signal=Rev Polarity) upon off-hook.	* 0x00 = Send SIGNAL message 0x01 = Do not send SIGNAL message
0x02	Send optional SIGNAL message (with steady signal=Normal Polarity) upon on-hook.	* 0x00 = Send SIGNAL message 0x01 = Do not send SIGNAL message
0x03	Deallocation option upon receiving Call Clearing message (send Channel Remove to BCC upon receiving on-hook, DISCONNECT or DISCONNECT COMPLETE)	* 0x00 = Send Channel Remove to BCC 0x01 = Do not send Channel Remove to BCC
0x04	Deallocation option upon sending DISCONNECT (LE side only)	* 0x00 = Send Channel Remove to BCC 0x01 = Do not send Channel Remove to BCC
0x05	Call Collision Priority	* 0x00 = Incoming call has priority 0x01 = Outgoing call has priority
0x06	Number of digits to receive before sending the SETUP Indication	Default = 10 digits
0x07	Request For Service Format	* 0x03 = V5 Formatted ICB
0x08	Specified the connected PPL state for PSTN	* 0x04
0x09	Config. Byte offset to IE Table. First Byte is number of IEs in table.	* 0x32
0x0A	Calling Party Control	0x00
0x0B	Send L3 ESTABLISH Indication to L5	* 0x00 = Do not send PPL Event Indication 0x01 = Send PPL Event Indication
0x0C	Reserved	
0x0D	Send L3 SIGNAL Indication to L5	* 0x00 = Do not send PPL Event Indication 0x01 = Send PPL Event Indication
0x0E	Send L3 STATUS Indication to L5	* 0x00 = Do not send PPL Event Indication 0x01 = Send PPL Event Indication
0x0F	Reserved	
0x10	Reserved	

Byte	Description	Values (* Default)
0x11	Reserved	
0x12	Send L3 PROTOCOL PARAMETER Indication to L5	* 0x00 = Do not send PPL Event Indication 0x01 = Send PPL Event Indication
0x14	Send SIGNAL messages with digit info for Called Party	* 0x00 = Do not send SIGNAL message 0x01 = Send SIGNAL message
0x15	Pulse Notification IE included in last SIGNAL message (Overlap Sending Option).	* 0x00 = Include 0x01 = Do not include
0x16	Stores Pulse Duration Type value Pulsed-Signal IE (FLASH Request)	* 0x00
0x17	Stores Suppression Ind. value Pulsed-Signal IE (FLASH Request)	* 0x00
0x18	Stores Acknowledge Request Ind. value for Pulsed-Signal IE (FLASH Request)	* 0x00
0x1A	Do not wait for SIGNAL messages with CLD (called) party number digits before sending SETUP Indication to L4	* 0x00 = Wait for digits 0x01 = Do not wait for digits
0x1B	Default Pulse Type to indicate a Switch Hook Flash	* 0x76 = Register Recall
0x20	Ringback Tone Option	* 0x00 = Do not play Ringback 0x01 = Send PPL Event Indication * 0x00 = Do not play Ringback 0x01 = Do play Ringback (make sure DSP card is configured with correct function type).

L3P/L5 Events

PPL Event Request (L5 to L3P)	PPL Event Indication (L3P to L5)
0x0000 Reserved 0x0002 SIGNAL Request 0x0003 Reserved 0x0004 Reserved 0x0005 Clear Request 0x0006 PROTOCOL PARAMETER Request	0x0000 Event Layer 3 Establish 0x0002 Reserved 0x0003 Event Layer 3 Signal 0x0004 Event Layer 3 Status 0x0005 Reserved 0x0006 Reserved 0x0007 Event Layer 3 Protocol parameter (only for AN side) 0x0014 Purge: PSTN initiated Purge for Userport when timeslot unknown 0x0015 Purge: PSTN returned to Idle state after purge 0x0016 L4 CONNECT REQ 0x0096 ERROR: No Channel Remove Acknowledge from BCC upon PURGE 0x0097 ERROR: No Disconnect Complete from L3 upon PURGE

L3P PSTN PPL Timers

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Layer 3P BCC Response Wait	8 seconds
0x02	Signal (On-hook/Normal Polarity)/Disconnect Complete Wait	4 seconds
0x03	Layer 3 Digit Wait Timer	4 seconds
0x04	Layer 4 Clear Wait Timer	4 seconds
0x05	Layer 3 Establish Acknowledge Wait	4 seconds

L3P BCC (0x92)

Purpose L3P BCC uses PPL configuration byte, event, and timer information.

L3P BCC Configuration Bytes

Byte	Description	Values (* Default)
0x01	Send Deallocate Request to L3 upon receiving Channel Remove message from PSTN	* 0x00 = Send Deallocate Request to L3 0x01 = Do not send Deallocate Request
0x02	Number of Allocation Request resends before giving up (send Channel Request reject to PSTN)	*0x00 =Default number of resends
0x03	Reserved	
0x04	Reserved	
0x05	Override the BCC Channels that are allocated and this can be set only from LE Side.	* 0x00 = Do not override 0x01 = Over

L5/L3 Events

PPL Event Indication (L5 to L3P)
0x0001 Layer 3 AUDIT Request (LE only) 0x0002 Layer 3 AUDIT Complete Indication

L3P BCC PPL Timers

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Layer 4 Router Response Wait	4 seconds
0x02	Layer 3 Audit Complete Wait	4 seconds
0x03	Layer 3 Audit Complete	4 seconds

L3P MGR (0x93)

Purpose L3P MGR uses PPL configuration byte, event, and timer information.

L3P MGR Configuration Bytes

Byte	Description	Values (* Default)
0x01	BCC Clear all calls or do not clear all calls when both C Channels are dead when switchover failure occurs.	*0x00 = BCC Clear All Calls 0x01 = BCC Do not clear all calls
0x02	PCM Encoding Format	0x01 = MU Law Encoded PCM *0x02 = A Law Encoded PCM 0x03 = Mixed Encoded PCM 0x01 = DS2 A Law Encoded PCM
0x03	Wait for Startup Request from host, when Layer Manager receives Startup Fail from System Management	*0x00 = Do not wait 0x01 = Wait
0x04	Type of Port Alignment **Important!** When this Config Byte is set to 2, the other side must be intelligent enough to send block indications to the ports that are locally blocked.	0x00 = Unblock all user ports, then block locally-blocked ports *0x01 = Perform accelerated port alignment procedure, then block locally-blocked ports 0x02 = Block locally-blocked ports
0x05	Send Reset Sequence Number during startup procedure	*0x00 = Do not send 0x01 = Send
0x06	Send all Status Indications to the host	*0x00 = Send only relevant Status Indications 0x01 = Send all Status Indications (used for diagnostics purposes only)
0x07	Enable Port Alignment Procedure	0x00 = Disabled (LE default) *0x01 = Enabled (AN default)
0x08	Link blocking action after startup	0x00 = Do not send block or unblock links (links will default to operation for the first startup) *0x01 = Send block link only for links that are locally blocked 0x02 = Send both unblock or block for all links depending on previous link status
0x09	Layer Manager Mode	*0x00 = Non-protective Mode 0x01 = Protective Mode

L3P/L5 Events

PPL Event Request (L5 to L3P)	PPL Event Indication (L3P to L5)
0x0000 Link Blocked Request Forced 0x0001 Link Deferred Block Request (AN Only) 0x0002 Link Unblock Request 0x0003 User Port Block Request Forced 0x0004 User Port Unblock Request 0x0005 C Channel Switchover Request 0x0006 V5 Statistics Query 0x0007 Link Status Request 0x0008 C Channel Status Request 0x0009 Interface Status Request 0x000A User Port Status Request 0x000B Start Restart Procedure 0x000C Reserved (Diagnostics: Disables LAPV for Link) 0x000D Reserved (Diagnostics: Enables LAPV for Link) 0x000E Link Non-Deferred Blocking Request (AN only) 0x000F User Port Deferred Block Request (AN only) 0x0010 Startup Request (Use this Event only when PPL Config. Byte #3 is set 0x0011 V5 Subscriber Query (Span/Channel and V5 ID/ User Port AIBs)	0x0001 Configuration OK 0x0002 Configuration Error 0x0003 Control Request OK 0x0004 Link MDU Event 0x0005 Control Error 0x0006 Status Error 0x0007 Statistics Error 0x0008 Management Port 0x0009 Management COM 0x000A Common Error 0x000B Bind Error 0x000C Event Message DEC 0x000D Event Port Error Message 0x000E Event L4 Request 0x000F Event Invalid State 0x0010 UI Event 0x0011 Interface Down 0x0012 Startup Event 0x0013 Restart Event 0x0014 Data Link Release Event 0x0015 Common 0x0016 Switchover Event 0x0017 Event Message Send 0x0018 Event Message Received 0x0019 L1 FSM Event 0x001A Port Error 0x001B DSFE 0x001C MPH 0x001D Timer Event 0x001E PV Layer Event 0x001F Event Timer Expire 0x0020 Waiting for L3P BCC Link Blocked Acknowledgment 0x0021 V5 Statistics 0x0022 Link Status 0x0023 C Channel Status 0x0024 Interface Status 0x0025 User Port Status 0x0026 No call active for timeslot 0x0027 Call is up for the timeslot (Contains appended V5 Subscriber ICB for the timeslot)

L3P MGR PPL Timers

When specifying new timer values, remember that all PPL Timers are specified in 10 ms increments. For example, a PPL timer for 4 seconds is specified as 4000 (decimal, 400 x 10 ms = 4 seconds), which is the hexadecimal value of (0x190).

Timer ID	Description	Value (default)
0x01	Link Blocked Acknowledge Wait	1 second

14 QSIG/PSS1

Purpose This chapter describes the Dialogic implementation and features of the Q Signaling/ Private Signaling System No. 1 (QSIG/PSS1) protocol.

QSIG/PSS1 Basic Call Signaling

Purpose

The CSP supports the QSIG/PSS1 global signaling and control standard for Private Integrated Network Exchange (PINX) applications, intended for use in private corporate ISDN networks. QSIG is a Euro-ISDN based protocol for digital Common Channel Signaling (CCS) and is used to build private networks using Virtual Private Networks (VPNs) or leased lines.

Q Signaling (QSIG), an ISDN based protocol, enables signaling between different voice communications platforms and equipment (nodes) in a multi-user environment. It is often referred to as an inter-PBX signaling system. It can also be deployed in a single-user environment.

Internationally, QSIG is also known as Private Signaling System No. 1 (PSS1).

Basic Call Signaling

The QSIG/PSS1 Basic Call Signaling provides signaling for establishing, maintaining and clearing a circuit-mode basic call at an interface between multiple PINXs. Layer 3 of QSIG, called QSIG Basic Call (BC), is a protocol that is symmetrical in nature. This means that interfaces from the network side and the user side are identical, so basic communication between multi-user nodes is transparent. Signaling between two PINX locations occurs when a call is set up and maintained in a multi-user environment. For example, if a caller from PINX A wants to make a call to a user in PINX B, QSIG BC ensures that certain mandatory features (such as call setup) are carried from one node to another.

The QSIG/PSS1 Basic Call Signaling is only supported on the ISDN Series 3 card. There is a limit of 32 D channels per ISDN Series 3 card and a system (node) limit of 64 D channels. This allows for a maximum of two ISDN Series 3 cards per chassis.

> **Important!** On a single ISDN Series 3 card there is a pool of 32 D channels that can be configured for QSIG, V5, or ISDN applications in any combination as long as the 32 channel limit is not exceeded.

The ISDN Configuration (CFG) task implemented on the ISDN Series 3 card supports the configuration for the QSIG variant. The *ISDN Interface Configure* (0x0060) API message also supports the configuration for the QSIG variant.

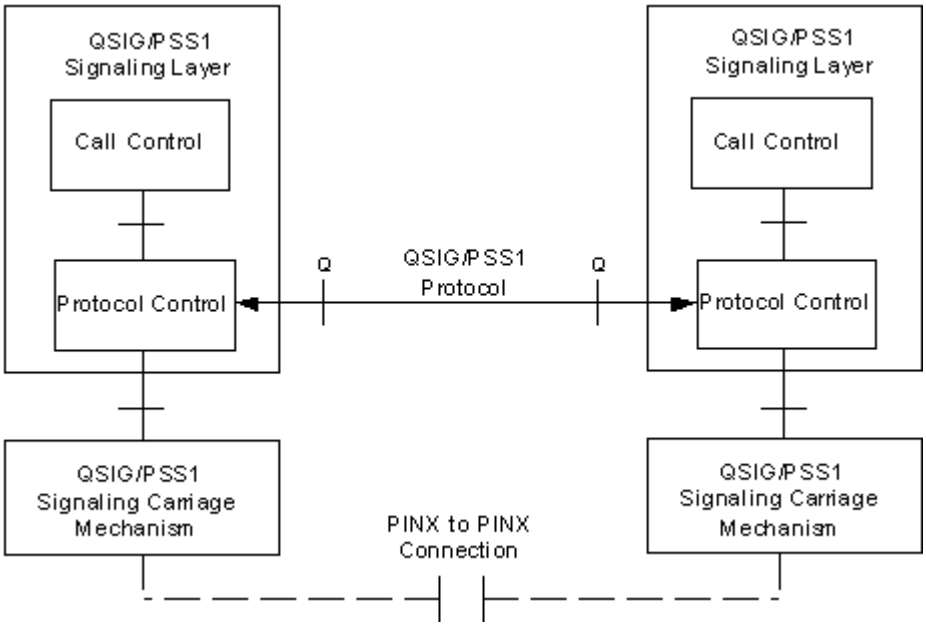
The ISDN IPRI task supports implementation of QSIG specific call control procedures. Segment messages are routed to the L3 Call Reference (0x08) component to the L3P Call Control (0x05) component and to the L4 Call Control Management (0x61) component. The segment message is sent to L4 Call Control Management (0x61) component in the M_MSG format, where it is then converted to Universal Protocol message format.

The L4 Call Control Management (0x61) component also supports the L4 CH SEGMENT *PPL Event Indication/Request* API messages.

In the figure below, the Protocol Control is implemented on the ISDN Series 3 card. The Call Control functionality will be implemented on the application running on the Matrix Host. In this way, depending on application running on the host, the switch can act as a Transit PINX, an Originating PINX, a Terminating PINX, an Incoming Gateway PINX, or an Outgoing Gateway PINX.

The signaling messages used to establish, maintain, and clear a circuit-mode Basic Call at an interface between the two PINXs are exchanged over a Signaling Carriage Mechanism (SCM) connection within the signaling channel of the Inter-PINX link.

Figure 14-1 QSIG/PSS1 Control Protocol Model



Message Segmentation and Reassembly

Message Segmentation and Reassembly procedures are required to support QSIG ISDN messaging. For detailed information, refer to the *QSIG/PSS1 Message Segmentation and Reassembly* section.

Licensing

QSIG is a licensed feature based on chassis ID. For detailed information, refer to the *QSIG/PSS1 Licensing* section.

Definitions

QSIG - Q interface Signalling protocol

PSS1 - Private Network Signalling System No. 1

PISN - Private ISDN

DSS1 - Digital Subscriber Signalling No. 1

ETSI - European Telecommunications Standards Institute

ISDN - Integrated Services Digital Network

ITU-T - International Telecommunication Union

PBX - Private Branch Exchange

PINX - Private ISDN Exchange

UPD - Universal protocol data format: A set of predefined data formats used by line card to communicate to L4 and above.

Compliance

QSIG BC is required of all users that claim QSIG compliance.

QSIG Generic Function (GF) Protocol

QSIG Generic Function (GF) protocol is not supported for this release.

QSIG/PSS1 Message Segmentation and Reassembly

Purpose

Message Segmentation and Reassembly procedures are required to support QSIG ISDN messaging. These procedures are implemented on the ISDN Series 3 card.

Message Byte Limits

The existing ISDN software supports messages in length of not greater than 260 bytes. The length of the ISDN messages supporting QSIG exceeds the 260 byte limit. These messages need to be segmented to fit into the 260 byte limit and later reassembled for use.

Since there is a limitation to the size of the messages which are routed between the layers (for example L3, L4, L5) the reassembled QSIG message cannot be sent directly to L5 (host). It must be divided again in parts of less than 250 bytes and sent to the L5 as one call processing API message, *Request for Service with Data* (containing the SEGMENT ICB), and the L4 CH *PPL Event Indication* API message.

From L5 to L4 the messages are sent in a similar way, one call processing API message, *Outseize Control* (containing the SEGMENT ICB), and the L4 CH *PPL Event Request* API message.

Message Segmentation and Reassembly Procedures

The processes of segmentation and reassembly can take place at the same time. The message segmentation and reassembly procedures are supported by the following:

The ISDN message (SEGMENT MESSAGE) and ISDN IE (Segmented IE) is supported.

The T14 timer is available for a reassembly procedures.

The following PPL event indications report segmentation procedure failures to L5 (host):

- Unexpected Segmented Message (0x0002)
- Segmentation Failure - Wrong or Unexpected Segmented Message (0x0003)

The following configuration bytes are available to enable or disable each of the above PPL event indications:

- Send Unexpected Segmented Message to host (0x23)
- Send Segmentation Failure - Timer Expired to host (0x24)

- Send Segmentation Failure - Wrong or Unexpected Segmented Message to host (0x25)

Important! Failures in reassembly procedures will not be reported to L5 (host).

QSIG Protocol

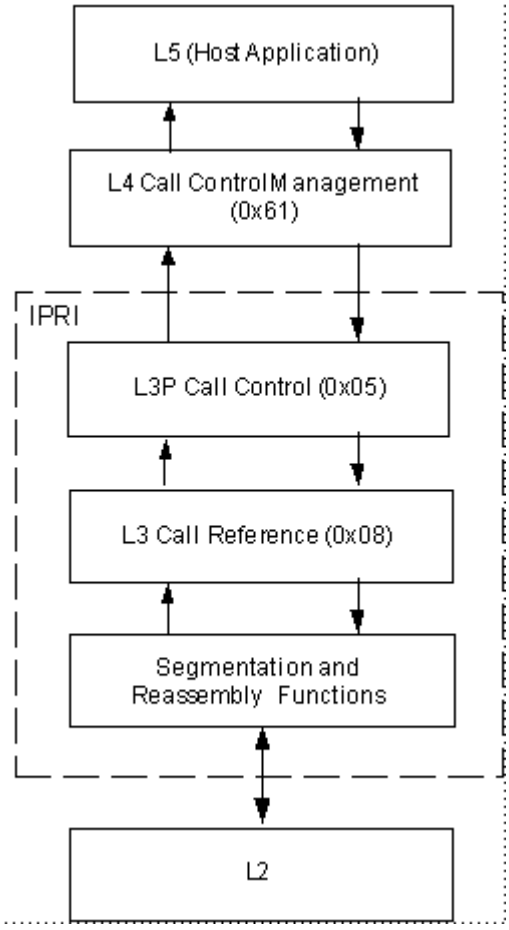
The QSIG Protocol, supported on the ISDN Series 3 card, provides the functions required to implement the Message Segmentation and Reassembly procedures which route messages to the L3 Call Reference (0x08) component which interfaces with L2.

- For the segmentation procedures, the call processing message and the segmented messages are received from L3 Call Reference (0x08) component.
- During reassembly, the call processing message and the segmented messages are sent to the L3 Call Reference (0x08) component.

Segment messages are routed to/from the L3 Call Reference (0x08) component, then to the L3P Call Control (0x05) component and then to the L4 Call Control Management (0x61) component. Segment messages sent to the L4 Call Control Management (0x61) component are in the Universal Protocol message format.

The L4 Call Control Management (0x61) component also supports the L4 CH SEGMENT *PPL Event Indication/Request* API messages.

Figure 14-2 Message Segmentation and Reassembly Routing



Reassembly

Reassembly procedures are initiated when the SEGMENT message is received from L2. The reassembly procedure cannot start on more than one B channel.

The SEGMENT messages will be buffered until receiving the last SEGMENT message for a given sequence (with “segment message to follow” parameter = 0).

When the last SEGMENT message arrives, the call processing message and all mandatory and optional parameters for this message will be extracted from the buffer (which contains all received SEGMENT Messages). At the end of the call processing message, the Segmented IE will be added. The message will then be sent to the L3 Call

Reference (0x08) component. All the others IEs, which are not recognized, as well as the Facility IE will be sent to the L3 Call Reference (0x08) component as an L3 Segment Message.

The extracted call processing message will be processed in the same manner as a normal call processing message. When it arrives at L5, the host application detects (from the segmented IE attached to the message), that there are SEGMENT messages coming after the call processing message. The SEGMENT messages will reach the host as L4 PPL Event Indication “SEGMENT message”.

If a failure condition or timeout occurs, the segmented messages buffered, so far will be discarded. A report for these events will not be sent to the host.

Segmentation

When received, a call processing message (containing a segmented IE) will be buffered in the segmentation buffer while waiting for all the remaining segmented messages to be sent by L5 as an L4 CH *PPL Event Request* API message.

The call processing message and segmented messages routed from the upper layers will be buffered and sent to L2 as N SEGMENTED Messages (N<=8).

If a failure condition or timeout occurs, the segmentation buffer will be emptied. A message from the L3 Call Reference (0x08) component indicating the failure, will invoke a PPL event indication. The PPL event indication will be sent from the L3 Call Reference (0x08) component because it must contain an AIB with a B channel identification. The following failures will be reported:

- When the Segmented Message comes from the upper layer not proceeding by a call processing message containing segmented IE: “Unexpected SEGMENT message”.
- When the timer expires: “Segmentation Failure - Timer Expired”
- When invalid SEGMENT Message comes or SEGMENT Message with a different Call Reference: “Segmentation Failure - Wrong or Unexpected SEGMENT Message”

SEGMENT Message Flow from L3 to L5 (Host)

When L3 receives all the SEGMENT messages, the messages are processed and sent to the L3 Call Reference (0x08) component as one call processing message and the rest as SEGMENT messages. All call processing messages (except the SETUP message) are routed to L5

(host) as L3P PPL Event indications (CONNECT, ALERTING, FACILITY, etc.). All call processing messages are routed to L5 (host) as follows:

- From L3 Call Reference (0x08) to L3P Call Control (0x05) and to L5 (host)

The SETUP message is routed to L5 (host) as an *RFS with Data* API message. The SETUP message is routed as follows:

- From L3 Call Reference (0x08) to L3P Call Control (0x05) to L4 Call Control Management (0x61) and to L5 (host)

The DISCONNECT message can be sent to L5 (host) in two ways: As a L3P PPL Event indication and/or as a *Channel Release Request* API message.

The following conditions must exist for sending the SEGMENT messages to L5 (host):

- The call processing message (SETUP, CONNECT, etc.) must reach the host first.
- All SEGMENT messages belonging to the call processing message must come in the same order as they were sent.
- The SEGMENT messages belonging to a certain call processing message must be sent to the host only if this call processing message is sent to the host. In other words, if the call processing message is not sent to the host due to a failure or configuration byte setting, then the SEGMENT messages belonging to it must be dropped out.
- The Segmented IE is replaced by the Segment ICB, so that the host application knows whether the API message is segmented or not.

All the SEGMENT messages and the SETUP message are routed to L5 the same way:

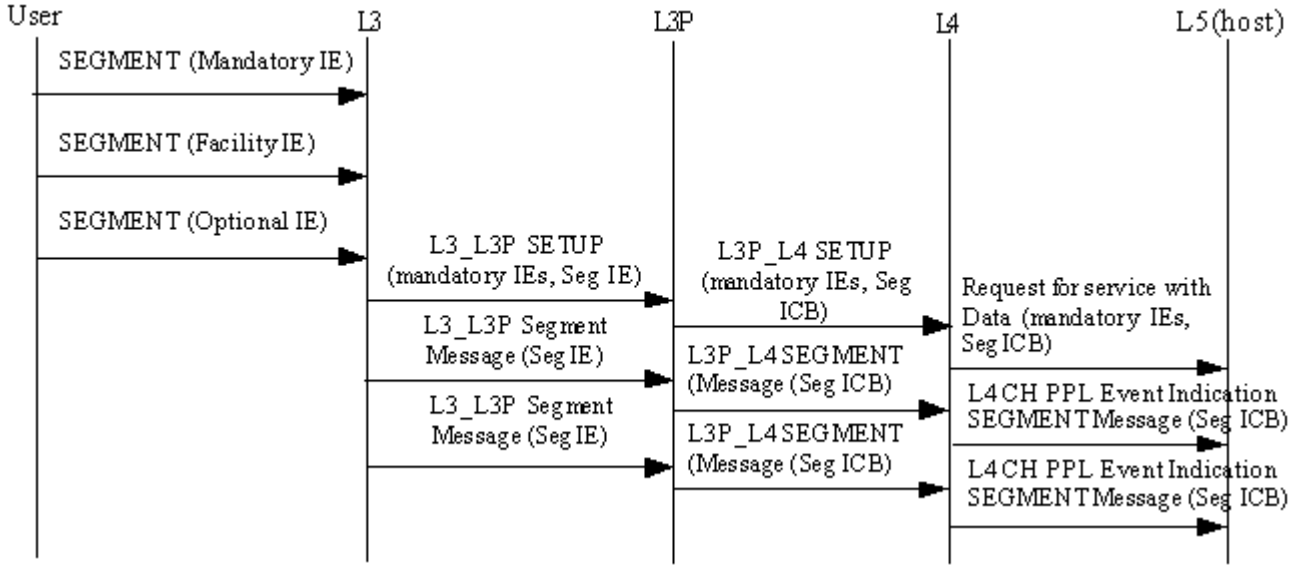
- From L3 Call Reference (0x08) to L3P CAll Control (0x05) to L4 Call Control Management (0x61) and to L5 (host)

This ensures that call processing messages will be the first to reach the host. SEGMENT messages will be sent as L4 CH PPL Event Indications.

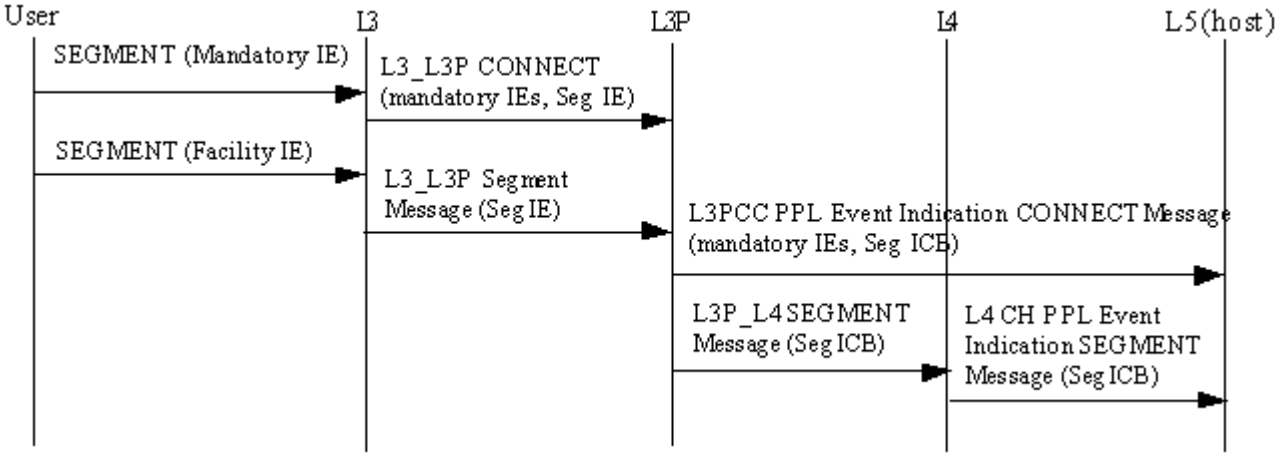
Segment Message Call Flows L3 to L5

Below are examples of SEGMENT Message call flows from L3 to L5:

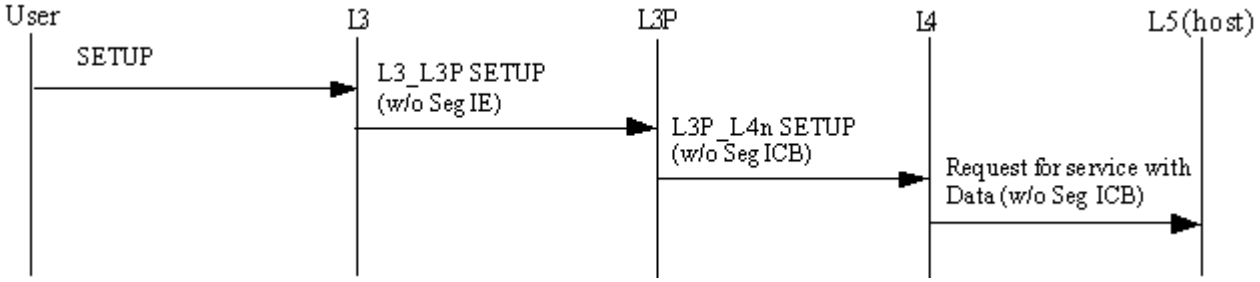
Three SEGMENT Messages which contain SETUP Message Inside



Two SEGMENT Messages which contain CONNECT Message Inside



L3 Receives other normal Call Processing Messages (for example, SETUP) from User

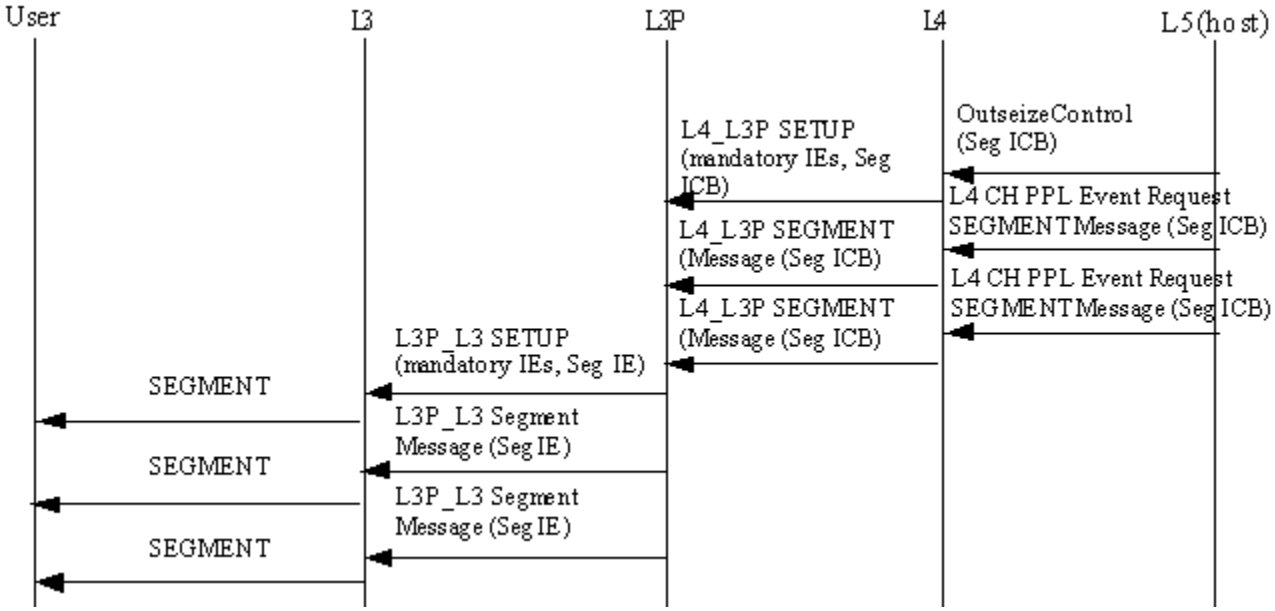


SEGMENT Message Call Flows from L5 (Host) to L3

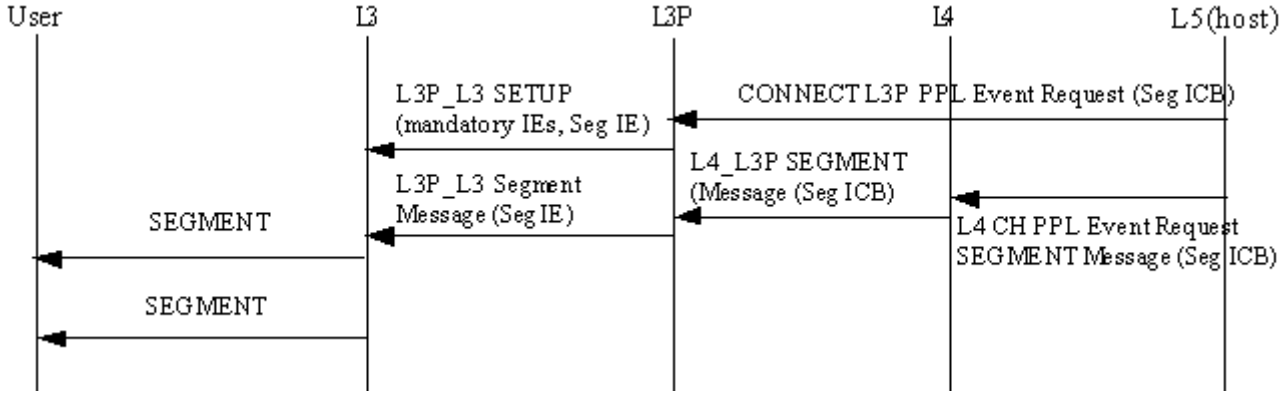
When the host application sends a message longer than 260 bytes, it will send the call processing message (*Outseize Control*, *Release Channel with Data*, etc.) containing the Segmented Message ICB or send the *PPL Event Request* message containing the Segmented Message ICB (in all other cases, CONNECT, ALERTING, FACILITY, etc.). The segmented messages are sent from L5 (host) as a L4 CH *PPL Event Request* message containing the Segmented Message ICB.

Below are examples of SEGMENT Message call flows from L5 to L3:

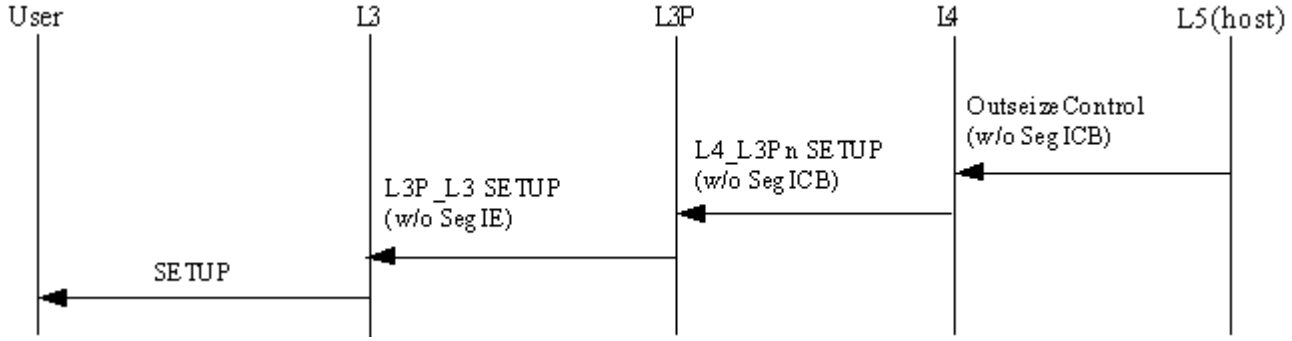
L5 Sends OUTSZ Messages followed by Two SEGMENT Messages



L5 Sends CONNECT PPL Event Request followed by One SEGMENT Message



L4 Sends Call Processing Message (for example, SETUP) without Segmented Message IE, with Message Length Less than 260 Bytes



QSIG/PSS1 Licensing

Licensing of QSIG/PSS1

The licensing QSIG/PSS1 Basic Call Signaling is based on the CSP chassis serial number.

Activating Licensing

To activate the QSIG/PSS1 feature, you must use a Product License Key, which Dialogic provides when you purchase the Product License. The key is unique and encrypted.

For details, refer to *Downloading License Keys to the CSP* in the *Licensing Overview* chapter in the *Developer's Guide: Overview* and the *Product License Download* message (0x0079) in the *API Reference*.

15 M3UA Implementation in the CSP

Overview

Purpose This chapter introduces the terminology and concepts regarding the M3UA functionality in the CSP.

Terminology Used

The following terminology is used in this document.

AS

Application Server. A logical entity serving a specific routing key. An example is a virtual switch element handling call processing for a unique range of PSTN trunks.

ASP

Application Server Process. An ASP serves as an active or backup process of an Application Server. An ASP can be configured to process signaling traffic with more than one Application Server.

Association

SCTP association providing transport for the delivery of the protocol data units for one or more interfaces.

HLR

Home location registry

IP

Internet Protocol

IPSP

IP Server Process is an instance of an IP based application. An IPSP is essentially the same as an ASP except that it uses M3UA in a point-to-point fashion. Conceptually, an IPSP does not use the services of a Signaling Gateway.

Layer Management

Nodal function in an SG or ASP that handles the inputs and outputs between the M3UA layer and a local management entity.

M3UA

MTP3 User Adaptation Layer

MGC

Media Gateway Controller

MTP

Message Transfer Part

Point Code

A unique code which identifies a network node in order for the SS7 network to route calls.

Routing Key

Describes a set of SS7 parameters and parameter values that uniquely define the range of signaling traffic to be handled by a particular Application Server.

Routing Contexts

The unique number assigned to each SG to AS logical connection. This number appears in the actual M3UA messages.

SCN

Switched Circuit Network

SCTP

Stream Control Transmission Protocol

SG

Signaling Gateway. An SG is a signaling agent that receives/sends switched circuit network signaling at the edge of the IP network. An SG appears to the SS7 network as an SS7 Signaling Point.

SGP

Signaling Gateway Process. A process instance of a Signaling Gateway that serves as an active, backup, load-sharing, or broadcast process of a Signaling Gateway.

SIGTRAN

Signaling Transport. A set of protocol standards defined by the International Engineering Task Force (IETF) to provide the architectural model of signaling transport over IP networks.

SMSC

Short message service center

SS7

Signaling System #7

Stream

SCTP stream is a unidirectional logical channel established from one SCTP end point to another. All messages are delivered in sequence except for unordered delivery service.

Message Transport Part Level 3 User Adaptation Layer (M3UA)

Overview

Signaling Transport (SIGTRAN) is a set of protocol standards defined by the International Engineering Task Force (IETF) to provide the architectural model of signaling transport over IP networks.

One of these standards is the MTP3 User Adaptation (M3UA) layer, which supports the transport of any SS7 MTP3 User Signaling over IP using the services of the Stream Control Transmission Protocol (SCTP).

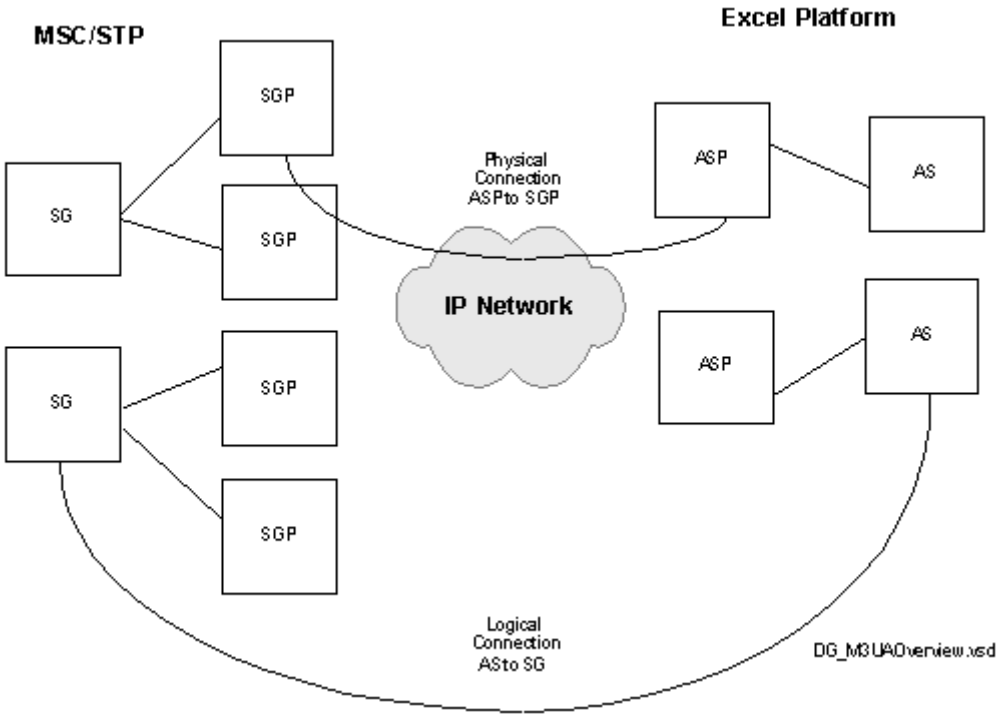
M3UA Network

In a network running M3UA, the Application Servers are the logical entity within the CSP. The Application Server Processes are the physical entities, which are physically connected to the Signaling Gateway Processes through the IP network.

The Signaling Gateways are the logical entities associated with the Signaling Gateway Processes. The two logical entities, Application Server and Signaling Gateway, are connected logically.

The figure below summarizes this type of network.

Figure 15-1 Physical and logical connections



Important Notes Regarding This Release

Overview

This section provides important information specific to this release of the M3UA software.

Performance

A single ASP supports 450 calls per second. A redundant ASP supports 400 calls per second.

ISUP and SSCP/TCAP Over M3UA Supported

This release supports ISDN User Part (ISUP) and SCCP/TCAP over the M3UA protocol.

Application Servers Have Unique Network Appearance

Each Application Server in the system must have a unique network appearance. The network appearance can have the same parameters (protocol type and network identity) but must have a different number.

Signaling Gateways and Signaling Gateway Processes

Signaling Gateways can use up to two Signaling Gateway Processes. A Signaling Gateway Process cannot serve more than one Signaling Gateway.

Load Share and Broadcast Options Not Supported

The load share and broadcast options for the traffic mode are not supported in this release. The Application Server side works in override mode only. This means that all the Mobile Subscriber Unit traffic sent from an Application Server to a specific Signaling Gateway will be sent by the same Application Server Process, even if the Application Server has more than one Application Server Process configured.

Licensing Information

M3UA is a licensable feature. For more information, refer to *Configuring M3UA Software (15-16)*.

IPSP and Remote SG Not Supported

This release of M3UA software does not support IP Server Process and Remote Signaling Gateway applications.

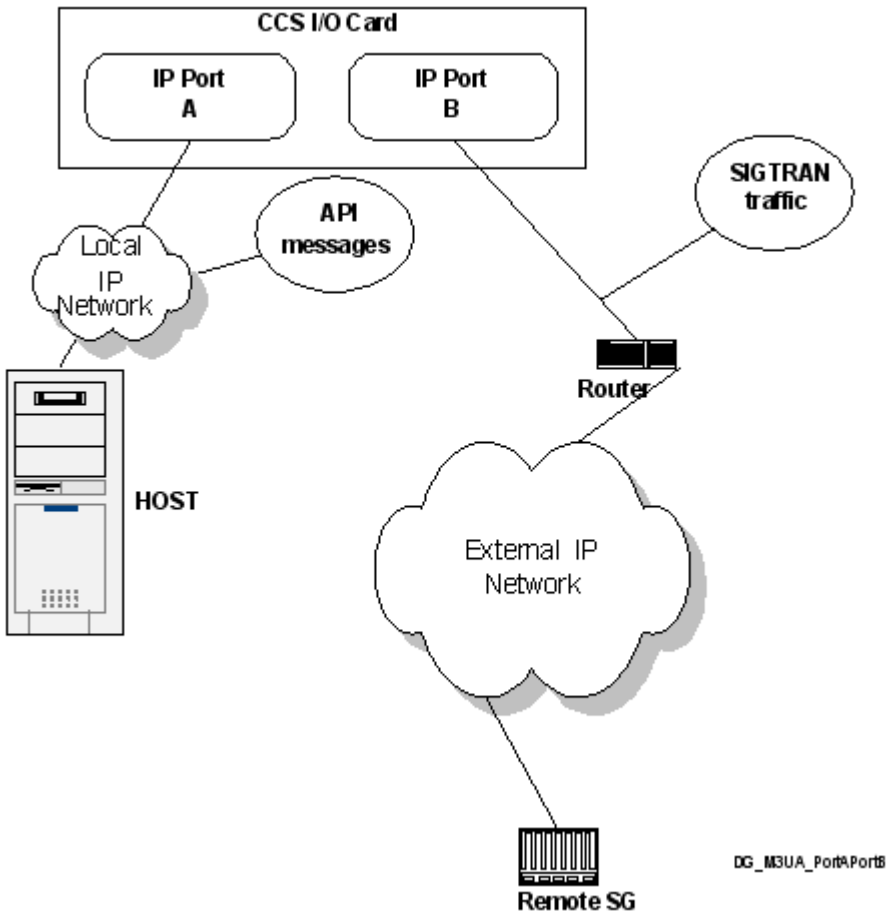
IP Network Configuration

Dialogic recommends the following IP network configuration:

- Only one gateway address is available, and it is always the first gateway address to appear in the IP configuration address.
- The system can have up to two Application Server Processes.
- The Ethernet A port on the CCS I/O card is for host traffic. The host and CSP should be in the same subnet, and no gateway address should be assigned to this port.

- The Ethernet B port on the CCS I/O card is for M3UA traffic. Configure the gateway address on this port so that it can communicate with other networks.

Figure 15-2 Separate Host and M3UA Traffic



M3UA Software Configuration

Overview

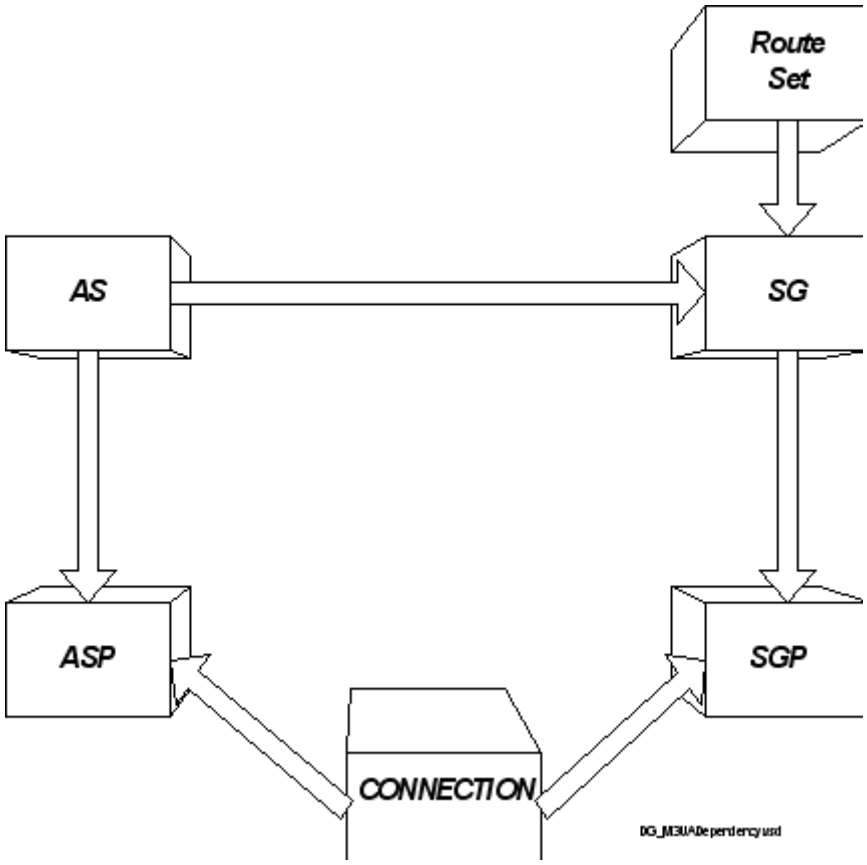
This section describes how to configure the M3UA software and then bring it into service.

The configuration procedures are presented in the order that you must complete them.

Dependencies

The figure below shows the dependencies of the M3UA objects which determines the sequence of configuration steps. This sequence is further explained in the *Overview of Configuration (15-10)*.

Figure 15-3 Dependencies between M3UA Objects



How to interpret this Figure

The object at the tail of the arrow depends on the object at the head of it. You cannot add an object before all of the objects that it depends on exist in the system.

You cannot remove an object before all of the objects depending on it are no longer configured in the system.

For example, Object Z depends on Object A. So Object A must exist in the system before you can add Object Z. If you need to remove these objects, you must remove Object Z before you remove Object A.

The objects contain the following information:

- Application Servers contain a list of Application Server Processes that the Application Server uses. Application Servers also contain a list of Signaling Gateways that the Application Server is communicating with.
- Signaling Gateways contain a list of Signaling Gateway Processes that the Signaling Gateway is using.
- The connection object points to the Application Server Process and Signaling Gateway Process that the connection object is connecting.

Overview of Configuration

Overview

The following steps provide an overview to configuring M3UA objects and bringing them in service. Each step is explained fully in *Configuring M3UA Software*. To remove M3UA objects reverse the order of these steps and use the Remove Command TLV instead of the Add Command TLV.

Prerequisites

The SS7 Card must have the IP Address configured (See the *IP Address Configure* message in the *API Reference.*)

Configuration Steps

All the steps below except steps 1 and 2 involve the *NGA Configure (0x0130)* message type:

1. Activate the M3UA Software.
2. Use the *SS7 Signaling Stack Configure 0x005C* message to configure the SS7 Stack to include the M3UA module.
3. Configure the M3UA Stack Parameter.
4. Add the Signaling Gateway Processes.
5. Add the Signaling Gateways.
6. Add the Route Sets.
7. Add the Application Server Process.
8. Add the Application Server. The Object ID of the Application Server must match the Stack ID.
9. Add the Connection.

Service Steps

Follow the steps below to bring the physical and logical connections in service.

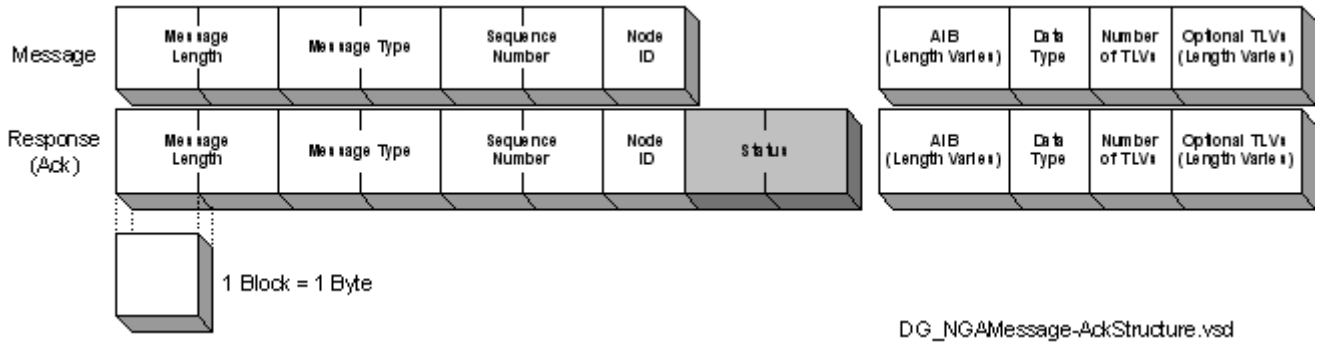
1. Bring the physical connection in service using the *NGA Service Configure (0x0160)* message type.
2. Bring the Application Server logical connection in service using the *NGA Service Configure (0x0160)* message type.

API Messages

This chapter includes tables that detail each field in the message you must provide. In addition, an example of each value is in the table. The header information for the message is not include. You can find that information in the *API Reference*.

The following is the basic structure for the API messages and API ACK messages used for M3UA software.

Figure 15-4 API Message Structure and Acknowledgment



Standard Configurations

Overview

This section introduces the following three standard M3UA configurations:

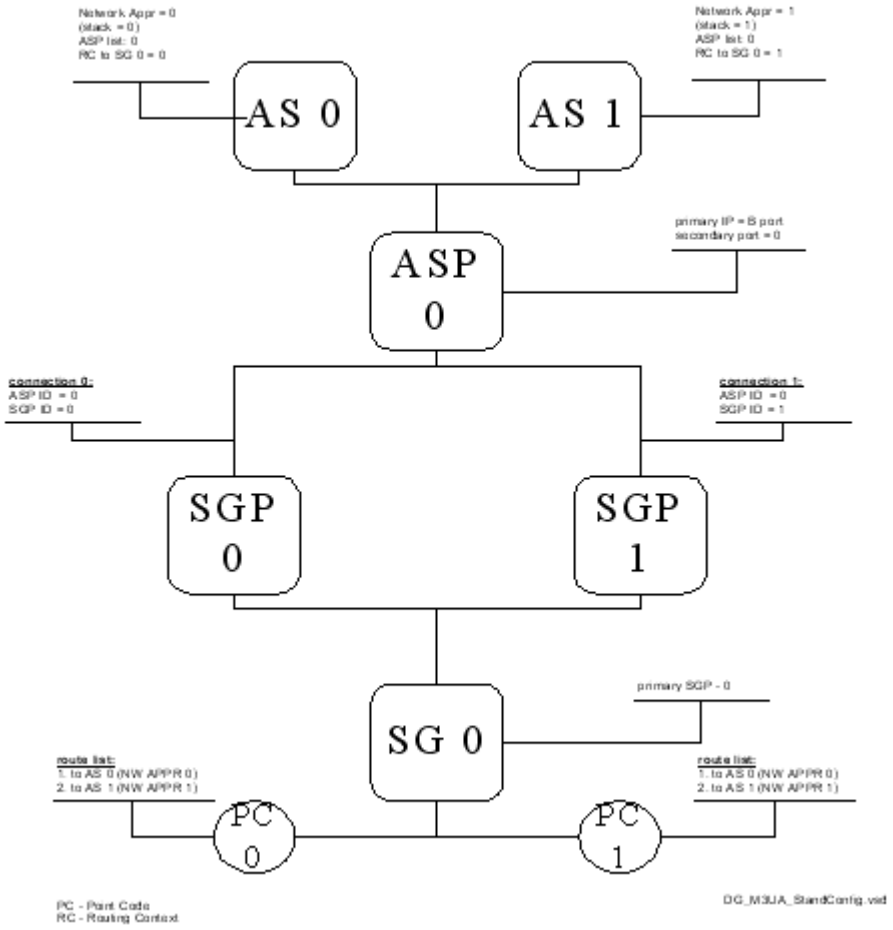
- Standalone Configuration - One SS7 Card
- Redundant Configuration- two Application Servers and two Application Server Processes
- Full Configuration - four Application Servers

Standalone Configuration

The standalone configuration typically uses one SS7 card. The following figure includes two Application Servers, one Signaling Gateway and one Signaling Gateway Process but you can configure the following:

- Up to four Application Servers
- Up to 10 Signaling Gateways each with one or two Signaling Gateways
- Up to 64 Route Sets

Figure 15-5 Standalone Configuration - One SS7 Card

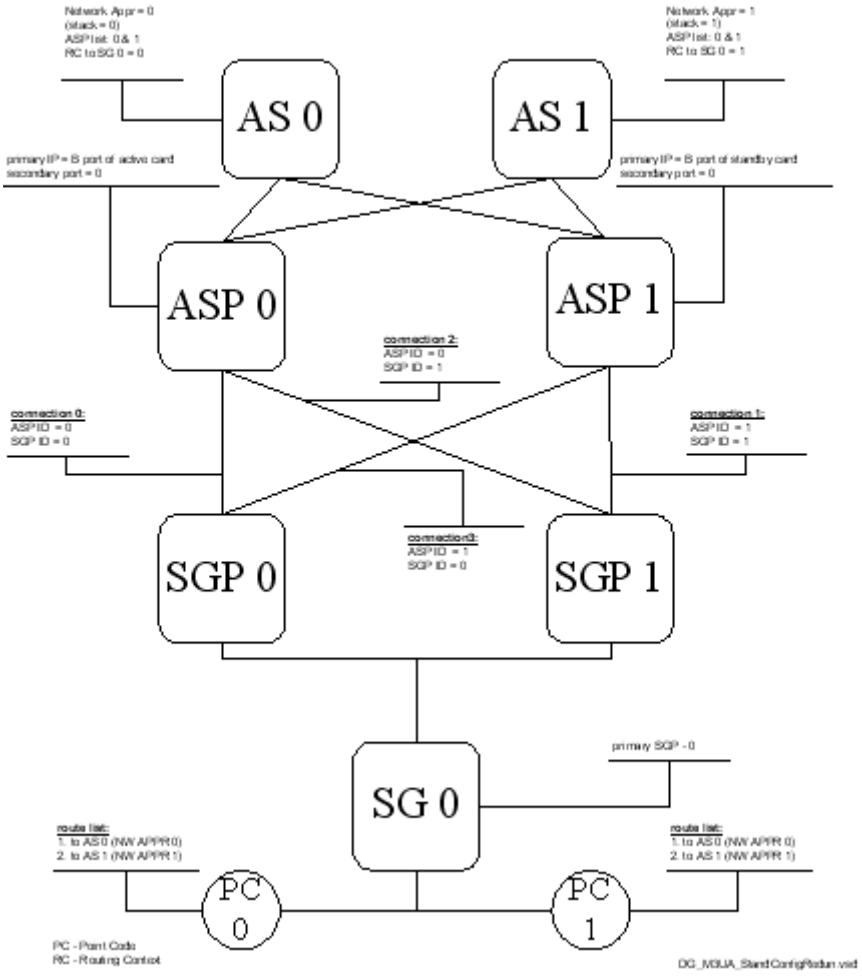


Redundant

A redundant configuration typically uses two SS7 cards in redundant mode. The figure below includes two Application Servers and one Signaling Gateway with two Signaling Gateway Processes. You can configure the following:

- Up to four Application Servers
- Up to 10 Signaling Gateways each with one or two Signaling Gateways
- Up to 64 Route Sets

Figure 15-6 Standard Configuration Redundancy

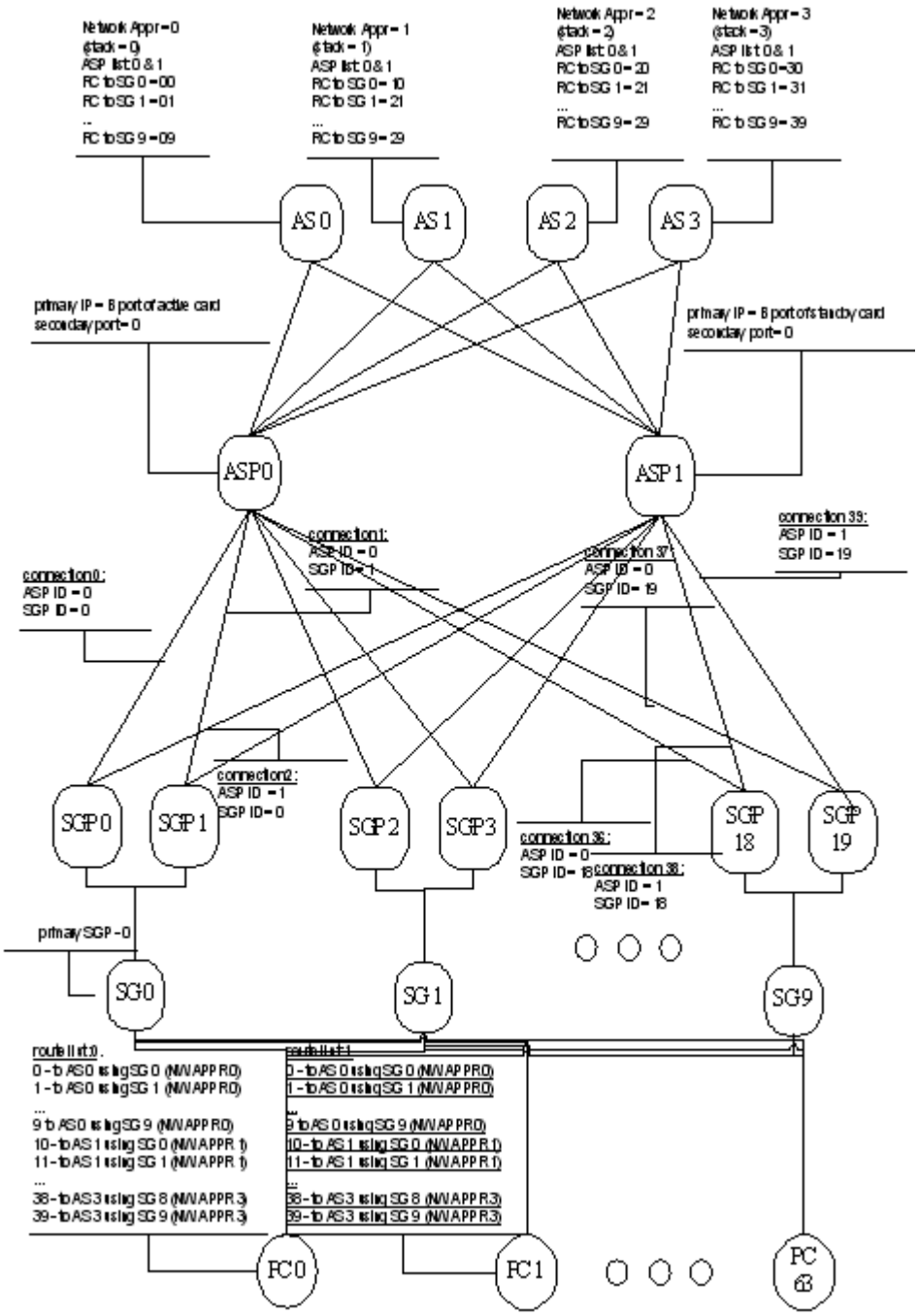


Full Configuration

The full configuration typically uses two SS7 cards in a redundant mode with the maximum number of objects of all types configured.

The following figure contains four Application Servers, 19 Signaling Gateway Processes, 10 Signaling Gateways, and 64 Point Codes.

Figure 15-7 Full Configuration



Configuring M3UA Software

Overview

Use the procedures in this section along with *API Reference* to configure M3UA software.

The procedures in this chapter includes an a description of the message and an example for each field. The header information is standard and is not included.

Activating M3UA Software

To activate any new feature, including M3UA Software, you must use a Product License Key, which Dialogic provides when you purchase the Product License. The key is unique and encrypted.

Follow the steps below:

1. Download the Product License to enable M3UA software. The M3UA license key is provided as a text file: usually *LICENSE.CFG* or *SERIALNO.CFG*.
2. Send the Product License Key in the Product License ICB (0x24), within the *Product License Download* message (0x0079). This message identifies both the key type and the license authorization code associated with a specific feature. The key type for M3UA is 0x3132. The CSP verifies the key, then activates the feature.
3. You must download the license key each time new system software is downloaded or when you cycle the power.

Configuring the SS7 Stack

Prerequisites

- The SS7 Card must have the IP Address configured (See the *IP Address Configure* message in the *API Reference*.)
- CCS Redundancy must be configured. (See the *CCS Redundancy Configure* message in the *API Reference*.)
- The Object ID of the Application Server must match the Stack ID.

To configure the SS7 Signaling Stack to include M3UA use the *SS7 Signaling Stack Configure 0x005C* message with the SS7 Slot AIB as follows:

Field	Example (hex)
AE - SS7 Slot	21
OPC	00FFFFFF
Number of Modules	03 (MTP, L3P, M3UA)
Module 1	01 - MTP
Module 1 Variant	00 - ANSI ’97
Module 2	03 - Layer 3 Plus
Module 2 Variant	00 - ANSI ’97
Module 3	08 - M3UA
Module 3 Variant	00 - ANSI ’97

Adding M3UA Protocol Parameters

When adding M3UA protocol parameters, you need to use maps if the CSP is working with different types of networks. For example the CSP might be working in an international and national network at the same time. For each network you need to assign a different map. If you are not using maps, enter 0 in the Map Number field.

Use the *NGA Configure (0x0130)* message type to create a map and assign values for standard and network identity to the map.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00

Field	Example (hex)
Object Identifier - SS7 Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran M3UA Stack Parameter	64 02 10 21
Object ID - -M3UA Stack ID Must be always 00 00	65 02 00 00
Data Type - Number of TLVs	00 02
Sigtran Command 0xE001 TLV	E0 01
Length	00 02
Add	00 00
Data TLV - 0x9041 M3UA Protocol Parameters	90 41
Length	00 15
Default Standard	00 00 00 01
Default Network Identity	00 00 00 00
Number of MAPs	01
MAPs Network Appearance	00 00 00 01
MAPs Network Standard	00 00 00 01
MAPs Network Identity	00 00 00 00

Modifying Protocol Parameters

Use the *NGA Configure (0x0130)* message type as described below. The example removes MAP 01.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20

Field	Example (hex)
Object Type - Sigtran M3UA Stack Parameter	64 02 10 21
Object ID - -M3UA Stack ID Must be always 00 00	65 02 00 00
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Modify	00 02
Sigtran Data TLV 0x9054	90 54
Length	00 11
MAP Modify Option 00 00 00 01 - Remove Map 00 00 00 00 - Add Map	00 00 00 01
Remove 01 Map	01
MAPs Network Appearance	00 00 00 01
MAPs Network Standard	00 00 00 01
MAPs Network Identity	00 00 00 00

Adding Signaling Gateway Process

You must configure all the Signaling Gateway Processes in the system and then configure the Signaling Gateways that use them. Note the following when adding Signaling Gateway Processes:

- A Signaling Gateway Process cannot be assigned to more than one Signaling Gateway.
- A Signaling Gateway can use up to two Signaling Gateway Processes.
- You can configure up to 20 Signaling Gateway Processes in the range 00-19.

Use the *NGA Configure (0x0130)* message type to add a Signaling Gateway Process.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Remote Signaling Gateway Process	64 02 10 29
Object ID - Remote Signaling Gateway Process	65 02 00 01
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Signaling Gateway Process Add Data - Data TLV 0x9045	90 45
Length	00 07
Port	0B 59
Number of IP Addresses	01
IP Addresses: 10.1.226.40	0A 01 E2 28

Removing Signaling Gateway Processes

You must remove all connections to the Signaling Gateway Process before you remove it. Use the *NGA Configure (0x0130)* message type described above but use the Remove (0x0001) value for the Sigtran Command TLV.

Adding Signaling Gateway

Once you configure all the Signaling Gateway Processes in the system, you can configure the Signaling Gateways that use them. Note the following when adding Signaling Gateways:

- A Signaling Gateway can use up to two Signaling Gateway Processes.
- You can configure up to 10 Signaling Gateways in the range 0-9.

- Use the *NGA Configure (0x0130)* message type to add a Signaling Gateway as indicated below.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Remote Signaling Gateway	64 02 10 27
Object ID - Remote Signaling Gateway	65 02 00 01
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Signaling Gateway Add Data - Data TLV 0x9044	90 44
Length	00 09
Traffic Mode: 01 Override; 02 Loadshare; 03 Broadcast	00 00 00 01
Primary SGP ID	00 01
Number of SGP	01
First SGP ID	00 01

Removing Signaling Gateways

You must remove all connections to a Signaling Gateway before you remove it.

The Signaling Gateways are represented in the system by the sum of the logical connections that lead to it. Refer to *Physical and logical connections (15-5)* for an overview of this concept.

Use the *NGA Configure (0x0130)* message type above but use the Remove (0x0001) value for the Sigtran Command TLV.

Adding Route Sets

Route Sets represent different ways to route a call through the network. Note the following when adding Route Sets:

- You can prioritize routes but you cannot load share.
- The signaling gateways associated with the route sets must exist before configuring the route sets.
- You can configure 64 route sets in the range 00 - 63. In the current release, the maximum number of routes in the system is 255.

Use the *NGA Configure (0x0130)* message type to add route sets as follows.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Route Set	64 02 10 2A
Object ID - Route Set	65 02 00 01
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Sigtran Data First- Data TLV 0x9052	90 52
Length	00 10
DPC	00 00 00 01

Field	Example (hex)
Number of Routes Up to 10 per Route Set (0-9). In current release, the maximum routes in the system is 255.	00 01
Route ID	00 01
Signaling Gateway ID	00 01
Network Appearance	00 00 00 01
Priority	00 01

Removing Route Sets

Use the *NGA Configure (0x0130)* message type as follows.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Route Set	64 02 10 2A
Object ID - Route Set	65 02 00 01
Data Type - Number of TLVs	00 01
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Remove	00 01

Modifying Route Sets

Use the *NGA Configure (0x0130)* message type as follows. The example removes route 1111.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA Protocol	64 02 10 20
Object Type - Sigtran Route Set	64 02 10 2A
Object ID - Route Set	65 02 00 01
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Modify	00 02
Data TLV - 0x9053 Modify Destination	90 53
Length	00 10
Modify Option 00 00 00 01 - Remove Route Set 00 00 00 00 - Add Route Set	00 00 00 01
Number of Routes Up to 10 per Route Set (0-9)	00 01
Route ID	00 01
Signaling Gateway ID	00 01
Network Appearance	00 00 00 01
Priority	00 01

Adding Application Server Process

You must configure the Application Server Processes before configuring the Application Servers that use them.

Note the following when adding Application Server Processes:

- Each Application Server Process should be assigned to a least one Application Server.
- You can configure two local Application Server Processes in the range 0-1.
- You can configure up to two IP Addresses per Application Server Process in the range 0-1.

Use the *NGA Configure (0x0130)* message type to add an Application Server Process as follows.

Field	Example (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Local Application Server Process	64 02 10 24
Object ID - Application Server Process	65 02 00 00
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Data TLV - 0x9043 Application Server Process	90 43
Length	00 08
Primary IP Address, B port of SS7: 135.119.48.82	87 77 30 52

Field	Example (hex)
Secondary IP Address A port of SS7 if not used	00 00 00 00

Removing Application Server Processes

To remove the Application Server Process, you must follow the steps below:

1. Set an Application Server Process to inactive for all Application Servers that use it.
2. Remove all connections using this Application Server Process.
3. Disconnect the Application Server Process from all Signaling Gateway Processes on the Signaling Gateway side.
4. Remove the Application Server Process from all the Application Servers.

Adding Application Server

You must configure the Application Server Processes and Signaling Gateways before configuring the Application Servers that use them. Use the *NGA Configure (0x0130)* message type to add an Application Server.

Note the following when adding Application Servers:

- You can configure up to four Application Servers in the range 0-3.
- Each Application Server in the system must have a unique network appearance. The network appearance can have the same parameters (protocol type and network identity) but must have a unique number.
- The Object ID of the Application Server must match the Stack ID. In this example, they are both 00 01.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Local Application Server	64 02 10 22
Object ID - Application Server	65 02 00 00

Field	Example Value (hex)
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Data TLV - 0x9042 Application Server	90 42
Length	00 25
Traffic Mode: 01 Override; 02 Loadshare; 03 Broadcast	00 00 00 01
Routing Key Type - 00=DPC	00 00 00 00
Routing Key Point Code	00 00 00 01
Routing Key Network Appearance	00 00 00 01
Routing Key Number of Ranges Up to 10 per AS in the range 0-9.	01
Routing Key Range OPC	00 00 00 01
Routing Key Range SIO	00
Routing Key SSN	00
Routing Key Low CIC	00 00
Routing Key High CIC	00 01
Number of Application Server Processes	01
ASP ID	00 00
Number of routing contexts Up to 10 per Application Server (0-9)	01
Routing Contexts	
Routing Contexts ID	00 00 00 01
Signaling Contexts SG ID	00 01

Removing Application Servers

You must stop all traffic to the Application Server by setting all the logical connections to inactive.

The inactive command should specify the Application Server that is going to be removed. This way the other Application Servers that are using the same Application Server Processes are not affected.

Adding a Connection

The Signaling Gateway Processes and Application Server Processes that will be connected should be configured before adding the connection.

You can configure 80 connections in the range 00-79.

Use the *NGA Configure (0x0130)* message type to add a connection.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Connection	64 02 10 2B
Object ID - Connection	65 02 00 01
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
Add	00 00
Data TLV - 0x9046 Connection	90 46
Length	00 04
Application Server Process ID	00 00
Signaling Gateway Process ID	00 01

Removing Connections

Removing a connection is a configuration action not a control action. You must first must stop all traffic on the connection by setting the Application Server that uses the logical connection to Out of Service.

You can send an inactive command to each Application Server that uses the Application Server Process or to all Application Servers using one command.

Use the *NGA Configure (0x0130)* message type described above but use the Remove (0x0001) value in the Sigtran Command TLV.

Bringing M3UA Objects into Service

Overview

Once the physical connection object and logical application server are configured, you bring them into service. You must perform the procedures in the order presented in this section.

Bringing the Physical Connection In Service

Use the *NGA Service Configure (0x0160)* to bring the connection identified by Object ID into service. Bringing the connection into service also brings the connected Application Server Process and Signaling Gateway Process in service.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Identifier - Stack ID	65 02 00 00
Object Type - Sigtran M3UA	64 02 10 20
Object Type - Sigtran Connection	64 02 10 2B
Object ID - Connection	65 02 00 01
Data Type - Number of TLVs	00 01
Sigtran Command TLV 0xE001	E0 01
Length	00 02
In Service	00 05

Taking the Physical Connection Out of Service

To take the Physical Connection out of service, use the *NGA Service Configure (0x0160)* message type as described above but use the Out of Service value (0x0006) for the Sigtran Command TLV.

Bringing the Logical Application Server in Service

Use the *NGA Service Configure (0x0160)* to bring the logical Application Server identified by the Object ID into service. Bringing the logical Application Server in service also bring the associated Signaling Gateway in service.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 00
Object Type - Sigtran M3UA Protocol	64 02 10 20
Object Type - Local Application Server	64 02 10 22
Object ID - Application Server	65 02 00 00
Data Type - Number of TLVs	00 02
Sigtran Command TLV 0xE001	E0 01
Length	00 02
In Service	00 05
Data TLV - Application Server Process Active	90 48
Length	00 04
Number of Logical Connections	00 01
Logical Connection ID	01 01

Taking the Logical Application Server Out of Service

To take the Logical Application Server out of service, use the *NGA Service Configure (0x0160)* message type as described above but use the Out of Service value (0x0006) for the Sigtran Command TLV. You do not need to use the fields after that TLV.

Host Notification

Purpose The M3UA software can notify the host about the following changes in an object's status:

- Application Server Status Change
- Application Sever Progress Status Change
- Connection Status Change

Application Server Status Change Notify

Overview

The M3UA software notifies the host of a status change to the Application Server with the *NGA State Notify (0x0163)* message type as follows. The example message notifies the host that Application Server 01 is in service. The logical connection 00 is out of service but the logical connection 01 is in service.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server	64 02 10 22
Object ID	65 02 00 01
0xE002 Service State	E0 02

Application Server Process Status Change Notify

Overview

The M3UA software notifies the host of a status change to the Application Server Process with the *NGA State Notify (0x0163)* message type as follows. The example message notifies the host that Application Server Process 01 is in service.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server Process	64 02 10 24
Object ID	65 02 00 01
0xE002 Service State	E0 02
In Service Out of Service	01

Connection Status Change Notify

Overview

The M3UA software notifies the host of a status change to the Connection with the *NGA State Notify (0x0163)* message type as follows. The example message notifies the host that connection 01 is out of service.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Connection	64 02 10 2B
Object ID	65 02 00 01
0xE002 Service State	E0 02
In Service Out of Service	00 00

Querying M3UA Data

Overview You can query the status of the specific object such as an Application Server or receive a list of all objects of that type:

- M3UA General Data
- Application Servers
- Signaling Gateways
- Signaling Gateway Processes
- Connections
- Route Sets
- Application Server Process

M3UA General Query

Overview

Use the *NGA Configure Query (0x0131)* message type. The response is TLV 0x904A.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - M3UA Stack Parameter	64 02 10 21
Object ID - Must be 0	65 02 00 00

Application Server Query

Overview

Use the *NGA Configure Query (0x0131)* message type. The responses are *0x9050 General Query Data* for a general query or *0x904B Application Server Query* for a query on a specific application server.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server	64 02 10 22
Object ID - Enter object ID or 0xFFFF to get the list of Application Servers	65 02 00 01

Application Server Process Query

Overview

Use the *NGA Configure Query (0x0131)* message type. The responses are *0x9050 General Query Data* for a general query or *0x904C Application Server Process Query* for a query on a specific application server process.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Process	64 02 10 24
Object ID - Enter object ID or 0xFFFF to get the list of Application Servers Processes	65 02 00 01

Signaling Gateway Query

Overview

Use the *NGA Configure Query (0x0131)* message type. The responses are *0x9050 General Query Data* for a general query or *0x904D Signaling Gateway Query* for a query on a specific Signaling Gateway.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Remote Signaling Gateway	64 02 10 27
Object ID - Enter object ID or 0xFFFF to get the list of Signaling Gateways	65 02 00 01

Signaling Gateway Process Query

Overview

Use the *NGA Configure Query (0x0131)*. The responses are TLVs *0x9050 General Query Data* for a general query or *0x904E Signaling Gateway Process Query* for a query on a specific signaling gateway process.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Remote Signaling Gateway Process	64 02 10 29
Object ID - Enter object ID or 0xFFFF to get the list of Signaling Gateway Processes	65 02 00 01

Connection Query

Overview

Use the *NGA Configure Query (0x0131)*. The responses are *0x9050 General Query Data* for a general query or *0x904F Connection Query* for a query on a specific connection.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Connection	64 02 10 2B
Object ID - Enter object ID or 0xFFFF to get the list of Connections	65 02 00 01

Route Set Query

Overview Use the *NGA Configure Query (0x0131)*. The responses are TLVs *0x9050 General Query Data* for a general query or the *0x9052 Route Set* for a query on a specific route set.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Route Sets	64 02 10 2A
Object ID - Enter object ID or 0xFFFF to get the list of Route Sets	65 02 00 01

Application Server Service State Query

Overview

You can query the service state which is the state in which the user put the Application Server.

Use the *NGA Service Query (0x0161)*. The response is the TLV *0x9051 Application Server State Query.*

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server	64 02 10 22
Object ID	65 02 00 01

Application Server Process Status Query

Overview You can query the actual state of an application server process.

Use the *NGA State Query (0x0162)*. The response is TLV *0xE002 Service State*.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server Process	64 02 10 24
Object ID	65 02 00 01

Connection Status Query

Overview Use the *NGA State Query (0x0162)*. The response is TLV *0xE002 Service State*.

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Connection	64 02 10 2B
Object ID	65 02 00 01

Application Server Status Query

Overview Use the *NGA State Query (0x0162)*. The response is TLV *0x9051 Application Server State Query.*

Field	Example Value (hex)
Address Method - Item	03
Functional Area - Signaling	67 02 00 01
Object Type - SS7 Stack	64 02 10 01
Object Identifier - Stack ID	65 02 00 01
Object Type - M3UA Protocol	64 02 10 20
Object Type - Local Application Server Process	64 02 10 24
Object ID	65 02 00 01

A Appendix

Purpose This appendix provides information on ISUP signaling messages, SSUTR2 signaling messages, signaling information, cause codes and message parameters. Descriptions of SS7 acronyms and ISUP messages are also provided.

SS7 Acronyms

Overview This section provides a list of SS7 acronyms used in this book.

ANSI	American National Standards Institute
APC	Adjacent Point Code
CCIS	Common Channel Interoffice Signaling
CCS7	Common Channel Signaling System 7
CIC	Circuit Identification Code
CPC	Call Processing Control
DCE	Data Communications Equipment
DPC	Destination Point Code
DTE	Data Terminal Equipment
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
L3P	Layer 3 Plus
MPC	Maintenance Process Control
MTP	Message Transfer Part

OPC	Originating Point Code
SCCP	Signaling Connection Control Part
SCP	Service Control Point
SLC	Signaling Link Code
SPRC	Signaling Procedure Control
SSP	Service Switching Point
SS7	Signaling System 7
STP	Signal Transfer Point

ISUP Signaling Messages

Message Code	ID	Description	Purpose
ACM	0x06	Address Complete Message	Sent in the backward direction indicating that all required data has been received and the call set-up is progressing
ANM	0x09	Answer Message	Sent in the backward direction to indicate that the called party has answered the call, may be used to trigger billing and call
BLA	0x15	Blocking Acknowledgment	Sent as a response to a BLO message indicating the identified circuit has
BLO [1]	0x13	Blocking	Sent for maintenance purposes to the exchange at the other end of the circuit to cause an engaged condition of that circuit (receive only)
CCL		Calling Party Clear	
CCR	0x11	Continuity Check Request	Sent to request a non-call-related continuity check on the identified circuit
CFN	0x2F	Confusion	Sent in response to any message (other than a confusion message) to indicate that all or part of a received message was unrecognized
CGB	0x18	Circuit Group Blocking	Sent for maintenance purposes to the exchange at the far end of a group of circuits to cause the engaged condition of those circuits
CGBA	0x1A	Circuit Group Blocking Acknowledgment	Sent as a response to a CGB indicating that the identified group of circuits has been blocked to outgoing traffic
CGU	0x19	Circuit Group Unblocking	Sent to cancel the engaged condition of a group of circuits caused by a previously sent CGB message.

CGUA	0x1B	Circuit Group Unblocking Acknowledgment	Sent in response to a CGU indicating the identified group of circuits are now in the idle state
CMC [3]	0x1D	Call Modification Complete	Message sent in either direction to indicate that the call modification request has been accepted
CMR[3]	0x1C	Call Modification Request	Message sent in either direction to request a change in the call characteristics
CMRJ [3]	0x1E	Call Modification Reject	Message sent in either direction to indicate that the call modification request has been rejected
CON [2]	0x0F	Connect	Sent in the backward direction indicating that all the address signals required for routing the call to the called party have been received and that the call has been answered
COT	0x05	Continuity	Sent in the forward direction to indicate the result of the completed continuity test
CPG	0x2C	Call Progress	Sent in either direction indicating that an event of significance to the originating or terminating access has occurred
CQM [3]	0x2A	Circuit Query Message	Sent on a routine or demand basis to request the exchange at the other end of a group of circuits give the state of the circuits within the specified range
CQR [3]	0x2B	Circuit Query Response	Sent in response to a CQM indicating the state of the previously identified group of circuits.
CRA [1]	0xE9	Circuit Reservation Acknowledgment	Sent in the backward direction in response to a CRM indicating that the circuit has been reserved for an incoming call
CRG [2]	0x31	Charge Information	Information sent in either direction for accounting and/or call charging purposes. national use only

CRM [1]	0xEA	Circuit Reservation Message	Sent in the forward direction only when interworking with exchange access MF signaling to reserve a circuit and initiate any required continuity check
CVR [1]	0xEB	Circuit Validation Response	Sent in response to a CVT to convey translation information for the indicated circuit
CVT [1]	0xEC	Circuit Validation Test	Sent on a routine or demand basis to request translation information for the identified circuit
DRS [2/3]	0x27	Delayed Release	Sent in either direction to indicate that the called or calling party has disconnected while the network is still holding the connection
En-block calls		Address digits are transmitted in one or more blocks.	In this form of signaling, address digits are transmitted in one or more blocks, and each block contains enough address information to enable the CSP to carry out forward routing.
EXM [1]	0xED	Exit Message	Sent in the backward direction from an outgoing gateway exchange to indicate that the call has successfully progressed to the adjacent network
FAA [2]	0x20	Facility Accept	Sent in response to a facility request message indicating that the requested facility has been invoked
FAC [3]	0x33	Facility Message	Sent in either direction at any phase of a call to request an action at another exchange. Also used to carry the results, error or rejection of a previously requested action
FAR [2]	0x1F	Facility Request	Sent from one exchange to another to request activation of a facility
FOT	0x08	Forward Transfer	Sent in the forward direction on semi-automatic calls when the operator wants the help of an operator at a distant exchange
FRJ [2]	0x21	Facility Reject	Sent in response to a facility request message indicating that the facility request has been rejected

GRA	0x29	Circuit Group Reset Acknowledgment	Sent in response to a GRS message to indicate the group of circuits have been realigned
GRS	0x17	Circuit Group Reset	Sent to align the state of a group of circuits, release any calls in progress remove any remotely blocked state.
IAM	0x01	Initial Address Message	Sent in the forward direction to initiate seizure of an outgoing circuit. May also request continuity test on identified circuit
IDR [2]	0x36	Identification Request	Sent in either direction to request an action regarding the malicious call identification supplementary service
IDS [2]	0x37	Identification Response	Sent in response to the identification request message
INF [3]	0x04	Information	Sent to convey additional call-related information which may have been requested in the inr message
INR [3]	0x03	Information Request	Sent by an exchange to request additional call related information
IRS		Identification	
LPA [3]		Loopback Acknowledge	
MCP		Message Compatibility Procedure	
MPM		Meter Pulse Message	
NRM [2]		Network Resource Management	Sent in order to modify network resources associated with a certain call, sent along an established path in any direction in any phase of a call
OLM [2/3]		Overload Message	Sent in the backward direction, on non-priority calls in response to an iam, to invoke temporary trunk blocking of the circuit concerned when the exchange generating the message is subject to load control
OPR		Operator	

PAM [3]	0x28	Pass Along Message	Sent in either direction to transfer information between two signaling points along the same signaling path as that used to establish a physical connection.
PCP		Parameter Compatibility procedure	
PDC		Propagation Delay Counter	
REL	0x0C	Release	Sent in either direction indicating that the circuit identified in the message is being released and the cause for the release
RES	0x0E	Resume	Sent in the backward direction to indicate re-answer from an interworking node or that a non-ISDN called party has gone off hook
RLC	0x10	Release Complete	Sent in either direction as a response to a release message or a reset circuit message to indicate that the circuit has been brought into the idle state
RSC	0x12	Reset Circuit	Sent when an exchange does not know the state of a particular circuit and wants to release any call in progress, remove any remotely blocked state and align states
SAM [2]	0x02	Subsequent Address Message	Sent in the forward direction to convey an additional segment of an overlength message after the IAM and before the ACM.
SGM [2]	0x38	Segmentation Message	Sent in either direction to convey an additional segment of an overlength message
SUS	0x0D	Suspend	Sent in the backward direction to indicate clear back from an interworking exchange or a non-ISDN called party has gone on hook
TTB		Temporary Trunk Blocking	
UBA	0x16	Unblocking Acknowledgment	Sent in response to a UBL indicating the identified circuit is now in the idle state

UBL	0x14	Unblocking	Sent to cancel the engaged condition of a circuit caused by a previously sent BLO message
UCIC [3]	0x2E	Unequipped CIC	Sent from one exchange to another when it receives a message that contains an unequipped circuit identification code
UPA [2]	0x35	User Part Available	Sent in either direction as a response to a user part test message, to indicate that the user part is available
UPT [2]	0x34	User Part Test	Sent in either direction to test the status of a user part marked as independent of call control messages
USR [2]	0x2D	User-Defined	Used for transport of user-to-user signaling independent of call control messages

1 Messages supported in ANSI only

2 Messages supported in ITU only

3 ITU - National use

ISUP Signaling Information

Overview This section includes definitions of ISUP messages.

Definitions The following messages can be modified using the *SS7 ISUP Message Format Configure* API message. Some messages are marked with a number which indicates the following:

[1]Specified for ANSI network

[2]Specified for ITU network

[3]ITU – for National use

Messages that are not marked with a number indicate use for an ANSI or ITU network.

Signaling Information	Definition
Access Delivery Indicator [2]	Information sent in the backward direction indicating that a SETUP message was generated at the destination access.
Access Transport	Information generated on the access side of a call and transferred transparently in either direction between the originating and terminating local exchanges. The information is of significance to both users and local exchanges.
Address Presentation Restricted Indicator	Information sent in either direction to indicate that the address information is not to be presented to a public network user, but can be passed to another public network.
Address Signal	An element of information in a network address. The address signal may indicate digit values 0 to 9, code 11 or code 12. One address signal value is reserved to indicate end of pulsing of the called party number (ST) and is not used.
Alarm Carrier Indicator [1]	Information sent in either direction indicating the type of carrier alarm handling.
Attendant Status [1]	An indication sent within the business group parameter signifying whether or not the business group information pertains to an attendant line.

Signaling Information	Definition
Automatic Congestion Level	Information optionally included in a Release Message and sent to the exchange at the other end of a circuit to indicate that a particular level of congestion exists at the sending exchange.
Backward Call Indicators	Information sent in the backward direction consisting of the charge indicator, called party's status indicator, called party's category indicator, end-to-end method indicator, interworking indicator, end-to-end information indicator, ISDN User Part indicator, holding indicator, ISDN access indicator, echo control device indicator, and SCCP method indication.
Business Group [1]	Information sent in either direction to indicate the identity of the Multi-Location Business Group associated with a number (e.g. the Calling Party Number), the identity of a Subgroup within that Multilocation Business Group and the Line Privileges allocated to the number.
Business Group Identifier [1]	Sent within the business group parameter to indicate the identity of the corresponding Multi-location Business Group.
Business Group Identifier Type [1]	Information sent within the business group parameter to indicate the type of business group identification used, e.g. multi-location business group identifier or interworking with private networks identifier.
Call Diversion Information [2]	Information sent in the backward direction indicating the redirection reason and the notification subscription option of the redirecting user.
Call Diversion May Occur Indicator [2]	Information sent in the backward direction indicating that call diversion may occur, depending on the response received (or lack thereof) from the called party.
Call History Information [2]	Information sent in the backward direction to indicate the accumulated propagation delay of a connection.
Call Identity [3]	Information sent in the call reference parameter indicating the identity of a call in a signaling point.

Signaling Information	Definition
Call Reference [3]	Circuit independent information identifying a particular call.
Called Party Number	Information sent in the forward direction to identify the called party and consisting of the odd/even indicator, nature of address indicator, numbering plan indicator, and address signals.
Called Party's Category Indicator	Information sent in the backward direction indicating the category of the called party, e.g. ordinary subscriber or pay phone.
Called Party's Status Indicator	Information sent in the backward direction indicating the status of the called party, e.g. subscriber free.
Calling Party Address Request Indicator [2/3]	Information sent in the backward direction indicating a request for the calling party address to be returned.
Calling Party Address Response Indicator [2/3]	Information sent in response to a request for the calling party address, indicating whether the requested address is included, not included, not available or incomplete.
Calling Party Number	Information sent in the forward direction to identify the calling party and consisting of the odd/even indictor, nature of address indicator, numbering plan indicator, address presentation restriction indicator, screening indicator, and address signals.
Calling Party Number Incomplete Indicator [2/3]	Information sent in the forward direction indicating that the complete calling party number is not included.
Calling Party's Category	Information sent in the forward direction indicating the category of the calling party, e.g. ordinary subscriber, test call.
Calling Party's Category Request Indicator [2/3]	Information sent in the backward direction indicating a request for the calling party's category to be returned.
Calling Party's Category Response Indicator [2/3]	Information sent in response to a request for the calling party's category, indicating whether or not the requested information is included in the response.

Signaling Information	Definition
Carrier Identification Code [1]	Information sent in the forward direction to the transit network indicating the transit network selected by the originating subscriber.
Carrier Selection Information [1]	Sent in the forward direction to indicate whether the calling user selected the transit network by pre-subscription or dialed input and if pre-subscribed whether or not the carrier identification code was also dialed.
Cause Indicators [1]	Information sent in either direction consisting of the coding standard, location, cause value and diagnostics. It indicates the reason for sending the message in which it is contained, e.g. the Release, Address complete or Confusion messages, and identifies the network in which the message originated, e.g. local network, transit network, remote local network.
Cause Value [2]	Information sent in either direction consisting of the coding standard, location, cause value and diagnostics. It indicates the reason for sending the message in which it is contained, e.g. the Release, Address complete or Confusion messages, and identifies the network in which the message originated, e.g. local network, transit network, remote local network.
Charge Indicator	Information sent in the backward direction indicating whether or not the call is chargeable.
Charge Information Request Indicator [2/3]	Information sent in either direction requesting charge information to be returned.
Charge Information Response Indicator [2/3]	Information sent in response to a request for charge information indicating whether or not the requested information is included.
Charge Number [1]	Information sent in either direction indicating the chargeable number for the call and consisting of the odd/even indicator, nature of address indicator, numbering plan indicator, and address signals.

Signaling Information	Definition
Circuit Assignment Map	Information identifying the map format and the circuits in a multi-rate connection. For the DS 1 map format it identifies the DS0 circuits constituting a NxDS0 connection. The circuit assignment map is used only for NxDS0 calls when non-contiguous time slot allocation is used.
Circuit Group Blocking Type Indicator [1]	Information sent in a Circuit Group Blocking (Unblocking) (Acknowledgment) Message indicating the blocking procedure used.
Circuit Group Carrier Indicator [1]	Information sent in either direction indicating if the circuit group carrier is analog, digital, or analog and digital.
Circuit Group Characteristics Indicator [1]	Information sent in response to a request for circuit validation, indicating the characteristics of the concerned circuit group in terms of carrier type, double seizing control, alarm handling and continuity check requirements.
Circuit Group Supervision Message Type Indicator	Information sent in either direction in a circuit group supervision message and consisting of the circuit group blocking type indicator.
Circuit Identification Code	Information identifying the physical path between a pair of exchanges.
Circuit Identification Name [1]	Information identifying the exchanges on which a circuit group terminates and the number of each trunk within that group.
Circuit State Indicator [3]	Information indicating the state of a circuit according to the sending exchange.
Circuit Validation Response Indicator	Information indicating the far-end results of a circuit validation test.
Closed User Group Call Indicator [2]	Information indicating whether or not the concerned call can be set up as a closed user group call and, if a closed user group call, whether or not outgoing access is allowed.
Closed User Group Interlock Code [2]	Information uniquely identifying a closed user group within a network.
Coding Standard	Information sent in association with a parameter (e.g. cause indicators) identifying the standard in which the parameter format is defined.

Signaling Information	Definition
Common Language Location Identification (CLLI) [1]	Information used for circuit validation to identify a switching office by town, state, and building subdivision. (COMMON LANGUAGE is a registered trademark and CLLI is a trademark of Bell Communications Research, Inc.)
Component Type [2/3]	There are four types of components that may be present in the Remote Operations parameter. The four Protocol Data Units (PDU) defined in Recommendation X.229 are used. (Invoke, Return Result, Return Error, Reject).
Connected Line Identity Request Indicator [2]	Information sent in the forward direction indicating a request for the connected party number to be returned.
Connected Number [2]	Information sent in the backward direction to identify the connected party.
Connection Request	Information sent in the forward direction on behalf of the Signaling Connection Control Part requesting the establishment of an end-to-end connection and consisting of the local reference, point code, protocol class, and credit.
Continuity Check Indicator	Information sent in the forward direction indicating whether or not a continuity check will be performed on the circuit(s) concerned or is being (has been) performed on a previous circuit in the connection.
Continuity Check Requirement Indicator [1]	Information sent in either direction indicating if and how often a continuity check is required for the circuit group.
Continuity Indicator	Information sent in the forward direction indicating whether or not the continuity check on the outgoing circuit was successful. A continuity check successful indication also implies continuity of the preceding circuits and successful verification of the path across the exchange with the specified degree of reliability.
Credit	Information sent in a connection request, indicating the window size requested by the signaling connection control part for an end-to-end connection.

Signaling Information	Definition
Diagnostic	Information sent in association with a cause value and which provides supplementary information about the reason for sending the message.
Discard Message Indicator [2]	Information sent to inform another node to discard the related message, due to compatibility reasons.
Discard Parameter Indicator [2]	Information sent to inform another node to discard the related parameter, due to compatibility reasons.
Double Seizing Control Indicator [1]	Information sent in either direction indicating the type of double seizing control applied to the circuit group.
Echo Control Device Indicator	Information sent in the forward direction indicating whether or not an outgoing half echo control device is included in the connection.
Egress Service [1]	Information sent in the forward direction by the first incoming exchange in the terminating exchange area, to indicate network specific attributes associated with the terminating exchange, e.g. the interexchange carrier, the point of interconnection, and type of terminating access service.
Encoding Scheme [2]	Information sent to indicate the coding type for the digit information, e.g. BCD-coded.
End of Optional Parameters [2]	The end of optional parameters field indicates that there are no more optional parameters in the message.
End-to-End Information Indicator	Information sent in either direction indicating whether or not the sending exchange has further call information available for end-to-end transmission.
End-to-End Method Indicator	Information sent in either direction indicating the available methods, if any, for end-to-end transfer of information.
Error Code [2/3]	The error code element contains the reason why an operation cannot be completed successfully. It is present only in a Return Error component. As with operations, errors may be local or global. These errors and associated parameters are defined in relevant supplementary service specifications.

Signaling Information	Definition
Event Indicator	Information sent in either direction to indicate the type of event that caused a Call Progress message to be sent. Event Information. Information sent in either direction consisting of the event indicator and event presentation restricted indicator that should be relayed to the access. Event Presentation Restricted Indicator. Information sent in either direction to indicate that the occurrence of the concerned event should not be reported to the user.
Event Information [1]	Information sent in either direction consisting of the event indicator and event presentation restriction indicator.
Event Presentation Restricted Indication [3]	Information sent in either direction to indicate that the occurrence of the concerned event should not be reported to the user.
Extension Indicator	Information indicating whether or not the associated octet has been extended.
Facility Indicator [2]	Information sent in facility related messages identifying the facility or facilities with which the message is concerned.
Feature Code [2/3]	Information sent in either direction to invoke, accept, or reject a specific action for a supplementary service.
Feature Code Indicator [1]	Information sent in either direction to invoke a specific feature operation at the terminating or originating CSP.
Filler [2]	A number of bits used to complete a partially used octet to full octet length.
Forward Call Indicators	Information sent in the forward direction consisting of the incoming international call indicator, end-to-end method indicator, interworking indicator, end-to-end information indicator, ISDN User Part indicator, ISDN User Part preference indicator, ISDN access indicator, and SCCP method indicator.
Generic Address [1]	Information in the form of an address pertaining to a supplementary service (e.g., dialed number, destination number) and including type of address, nature of address and numbering plan indications.

Signaling Information	Definition
Generic Digits [3]	Information in the form of digits pertaining to a supplementary service, (e.g. account code, authorization code, private network class mark) and including type of digits and encoding method indicators.
Generic Name	Information sent in the forward direction containing specific name-related information.
Generic Notification [2]	Information sent in either direction intended to provide supplementary service notification to a user.
Generic Number [2]	A number information sent in either direction to enhance network operation or for supplementary services.
Generic Reference (reserved) [2]	For further study.
Hold Provided Indicator [2/3]	Information sent in the forward direction indicating that the connection will be held after the calling or called party has attempted release.
Holding Indicator [2/3]	Information sent in the backward direction indicating that holding of the connection is requested.
Hop Counter (ITU reserved for further study)	Information sent in the forward direction to minimize the impact of lAM looping. The initial count determines the maximum number of contiguous SS7 interexchange circuits that are allowed to complete the call, assuming all subsequent intermediate exchanges decrement the hop counter.
In-band Information Indicator	Information sent in the backward direction to indicate that in-band information, e.g. a tone or announcement, is present on the circuit path.
Incoming Half Echo Control Device Request Indicator [2]	Information sent to request the activation or deactivation of an incoming half echo control device.
Incoming Half Echo Control Device Response Indicator [2]	Information sent to inform whether an incoming half echo control device has been included or not.
Incoming International Call Indicator [1]	Information sent in the forward direction indicating whether the call is an incoming international or an incoming national call.

Signaling Information	Definition
Information Indicators	Information sent in either direction consisting of the calling party address response indicator, connected address response indicator, calling party's category response indicator, charge information response indicator, and solicited information indicator.
Information Request Indicators	Information sent in either direction consisting of the calling party address request indicator, connected address request indicator, calling party's category request indicator, charge information request indicator, malicious call identification request indicator, and holding indicator.
Information Transfer Capability [1]	Information sent in the forward direction indicating the type of transmission medium required for the connection.
Information Transfer Rate [1]	Information sent in the forward direction indicating the information transfer rate
Instruction Indicator [2]	Information indicating the reactions to be taken if an unrecognized message or unrecognized parameter is received.
Internal Network Number [2]	Information sent to the destination exchange for specific numbers, e.g. roaming numbers indicating whether or not the network generates the number contained in the parameter.
Interworking Indicator	Information sent in either direction indicating whether or not Signaling System No. 7 is used in all parts of the network connection.
Invoke Identification [2/3]	An Invoke ID is used as a reference number to identify uniquely an operation invocation. It is present in the invoke component and in any reply to the invoke enabling the reply to be correlated with the Invoke.
ISDN Access Indicator	Information sent in either direction indicating whether or not the access signaling protocol is ISDN.
ISDN User Part Indicator	Information sent in either direction to indicate that the ISDN User Part is used in all parts of the network connection.

Signaling Information	Definition
ISDN User Part Preference Indicator	Information sent in the forward direction indicating whether or not the ISDN User Part is required or preferred in all parts of the network connection. Jurisdiction Information sent in the forward direction indicating the geographic origination of the call.
Jurisdiction Information [1]	Information sent in the forward direction indicating the geographic origination of the call.
Length of Network Identification [2/3])	Information sent in the network specific facility parameter, to indicate the length in octets of the network identification.
Length of Reference Indicator [2] (for further study)	Information sent in the generic reference parameter, to indicate the length in octets of the reference.
Line Privileges [1]	Information sent within the business group parameter to indicate the privileges of the user identified by the corresponding number,
Line Privileges Information Indicator [1]	Information sent within the business group parameter signifying whether the indicated line privileges are standard or customer defined.
Linked Identification [2/3]	A linked ID is included in an Invoke component by a node when it responds to an operation invocation with a linked operation invocation. The node receiving the linked ID uses it for correlation purposes in the same way that it uses the invoke ID in a Return Result, Return Error, or Reject.
Local Reference	Information sent in the connection request, indicating the local reference allocated by the signaling connection control part to an end-to-end connection.
Location [2]	Information sent in either direction indicating where an event was generated.
Location Number [2]	Information sent to indicate the location of a user in the term of an E.164 number.
Look Ahead For Busy [1]	Information sent within the precedence parameter indicating whether the look-ahead for busy option is allowed or whether the path has been reserved.

Signaling Information	Definition
Look For Busy (LFB) [2]	Information sent in the forward direction to indicate whether LFB option is allowed or if the path for the call is reserved.
Malicious Call Identification Response Indicator [2/3]	Information sent in the forward direction indicating whether the malicious call identification has been provided or not.
MCID Request Indicator [2]	Information sent in the backward direction to request the identity of the calling party for the purpose of malicious call identification.
MCID Response Indicator [2]	Information sent in the forward direction to respond to a MCID request and indicating whether or not the MCID information is available.
Message Compatibility Information Parameter [2]	Information sent in either direction indicating how an exchange should react in case this message is unrecognized.
MLPP Service Domain	Information sent in the precedence parameter identifying the network resources to which the Multi-level Precedence and Preemption supplementary service is applicable for the call.
MLPP User Indicator [2]	Information sent in the backward direction to indicate that the called user is an MLPP user.
National/International Call Indicator [2]	Information sent in the forward direction indicating in the destination national network whether the call has to be treated as an international call or as a national call.
Nature of Address Indicator	Information sent in association with an address indicating the nature of that address, e.g. ISDN international number, ISDN national significant number, or ISDN subscriber number.
Nature of Connection Indicators [1]	Information sent in the forward direction consisting of the satellite indicator, continuity check indicator, and echo control device indicator.
Network Discard Indicator [2]	This indicator indicates that the network has discarded user-to-user information included in the call control message.
Network Identification [1]	Information sent in the forward direction indicating the identity of the transit network.
Network Identification [2/3]	Information sent to identify a network.

Signaling Information	Definition
Network Identification Plan [1]	Information sent in association with the transit network selection information to identify the type of network identification used. Network Transport A parameter sent in either direction for the purpose of transporting other ISDN User Part parameters transparently across transit exchanges.
Network Identification Plan [2/3]	Information sent to indicate the identification plan for identifying the network, e.g. X.121 or E.212, (DNIC or MNIC).
Network Identity [2/3]	Information sent to identify the network that administrates the supplementary service.
Network Specific Facilities [2/3]	Service related information transparently transferred in either direction between the local exchange and the identified network, which contracts the service. The information is significant to both user and the identified network.
Network Transport	A Parameter sent in either direction for the purpose of transporting other ISDN User Part parameters transparently across transit exchanges.
Notification Indicator [1]	Information Sent in either direction intended to provide supplementary service related notification to a user.
Notification Subscription Option [2]	Information sent in the backward direction indicating that the diversion with or without redirection number can be presented to the calling user.
Number Incomplete Indicator [2]	Information sent in the generic number parameter to indicate whether the delivered number is complete or incomplete.
Numbering Plan Indicator	Information sent in association with an address indicating the numbering plan used for that address (e.g., ISDN numbering plan).
Odd/Even Indicator	Information sent in association with an address, indicating whether the number of address signals contained in the address is even or odd.

Signaling Information	Definition
Operation Code [2/3]	The operation code element indicates the precise operation to be invoked, and is present in an Invoke component type. It is also present in the Return Result component if the result contains parameters. The operation may be a local or global operation. A local operation can be used in one ASE only. The same global operation can be used in several different ASEs. The actual operation codes, the definition of the operations and their associated parameters, are defined in relevant supplementary service specifications.
Operator Services Information [1]	Information sent in the forward direction between operator services entities primarily identifying charging and service type options.
Optional Backward Call Indicator [1]	Information sent in the backward direction consisting of the in-band information indicator, the call forwarding may occur indicator and the user-network interaction indicator.
Original Called Number	Information sent in the forward direction to indicate, in the case of call redirection (e.g., call forwarding), the number of the user who initiated the initial redirection.
Original Redirecting Reason [1]	Information sent in the forward direction indicating the forwarding condition for the original redirection.
Original Redirection Reason [2]	Information sent in either direction indicating the reason why the call was originally redirected.
Originating Line Information [1]	Information sent in the forward direction, indicating a toll class of service for the call.
Origination ISC Point Code [2]	Information sent in the initial address message of an international call, indicating the point code of the originating ISC.
Outgoing Half Echo Control Device Request Indicator [2]	Information sent to request the activation or deactivation of an outgoing half echo control device.
Outgoing Half Echo Control Device Response Indicator [2]	Information sent to inform whether an outgoing half echo control device has been included or not.
Outgoing Trunk Group Number [1]	Information sent in the backward direction indicating the trunk group selected at an outgoing gateway. For intra-network use only.

Signaling Information	Definition
Parameter Compatibility Information [2]	Information sent in either direction indicating how an exchange should react in case the parameter is unrecognized.
Party Selector [1]	Information sent within a business group parameter to indicate the type of number to which the business group information applies.
Pass On Not Possible indicator [2]	Information sent to inform another node on what action to take if “pass on” was requested due to compatibility reason but “pass on” was not possible due to interworking with pre-ISUP 1992 signaling.
Point Code	Information sent in the call reference parameter indicating the code of the signaling point in which the call identity allocated to the call reference is relevant.
Precedence [1]	Information sent in the forward direction in association with the invocation of the Multi-Level Precedence and Preemption (MIYP) supplementary service, consisting of the look-ahead for busy indication, the precedence level, network identity, and the MI2P service domain.
Precedence Level [1]	Information sent in the precedence parameter indicating the precedence level of the call.
Precedence Level [2]	Information sent in the forward direction to indicate the priority of the call.
Problem Code [2]	The problem code element contains the reason for the rejection of a component and one such element is present in a Reject component. Four problem code elements are defined (General Problem, Invoke Problem, Return Result Problem, and Return Error Problem.)
Propagation Delay Counter [2]	Information sent in the forward direction to indicate the propagation delay of a connection. This information is accumulated while the parameter is transferred through the network. The propagation delay information is represented by a counter counting in integer multiples of 1 ms.

Signaling Information	Definition
Protocol Class	Information sent in the connection request parameter indicating the protocol class requested by the signaling connection control part for the end-to-end connection
Protocol Control Indicator [1]	Information consisting of the end-to-end method indicator, the interworking indicator, the end-to-end information indicator and the ISDN User Part indicator. The protocol control indicator is contained in both the forward and backward call indicators parameter fields and describes the signaling capabilities within the network connection.
Protocol Control Indicator [2]	Information consisting of the end-to-end method indicator, the interworking indicator, the SCCP Method indicator and the ISDN User Part indicator. The protocol control indicator is contained in both the forward and backward call indicators parameter fields and describes the signaling capabilities within the network connection.
Protocol Profile [2/3]	Information sent in either direction to indicate the protocol used in the Remote Operations parameter.
Range	Information sent in a circuit group supervision message (e.g. Circuit Group Blocking) to indicate the range of circuits affected by the action in the message.
Range and Status [1]	Information sent in either direction in circuit group supervision messages consisting of the range and status.
Redirecting Indicator [2]	Information sent in either direction indicating whether the call has been diverted or rerouted and whether or not presentation of redirection information to the calling party is restricted.
Redirecting Number	Information sent in the forward direction indicating the number from which the call was last redirected and consisting of the nature of address indicator, numbering plan indicator, address presentation restriction indicator, and address information (signals).
Redirecting Reason	Information sent in the forward direction indicating the forwarding condition for the last redirection.

Signaling Information	Definition
Redirection Counter	Information sent in the forward direction indicating the number of forwardings undergone.
Redirection Indicator [2]	Information sent to indicate whether the call has undergone diversion or rerouting. It also contains information about presentation restrictions.
Redirection Information	Information sent in the forward direction consisting of the original redirecting reason redirection counter and redirecting reason.
Redirection Number [2]	Information sent in the backward direction indicating the number towards which the call must be rerouted or has been forwarded.
Redirection Number Restriction Indicator [2]	Information sent in the backward direction indicating whether the diverted-to user allows the presentation of his number.
Redirection Reason [2]	Information sent in the call diversion information parameter and the redirection information parameter to indicate the reason for the redirection.
Reference n^{th} Octet [2] (for further study)	Information sent in the generic reference parameter, expressing the reference number of the context given by the entity, which handles and provides the service.
Reference Qualifier Indicator [2] (for further study)	Information sent in the generic reference parameter, identifying the context that handles and provides service.
Release Call Indicator [2]	Information sent to inform another node to release the call or not, by compatibility reasons, if the related message or parameter is unrecognized.
Remote Operations [3]	The remote operations parameter is used to indicate the invocation of a supplementary service identified by an operation value and also carry the result or error indications depending on the outcome of the operation.
Routing Label	Information provided to the message transfer part for the purpose of message routing.
Satellite Indicator	Information sent in the forward direction indicating the number of satellite circuits in the connection.

Signaling Information	Definition
SCCP Method Indicator	Information sent in the forward direction indicating the available SCCP methods, if any, for end-to-end transfer of information.
Screening Indicator	Information sent in association with a number indicating whether the number was network or user provided, and if user provided, whether the network views the number as correctly identifying the user.
Send Notification Indicator [2]	Information sent to inform another node to send notification, due to compatibility reason, if the related message or parameter is unrecognized.
Sequence [2/3]	The sequence is an ordered set.
Service Activation [1]	Information sent in either direction to indicate the invocation of one or more supplementary services.
Service Activation Parameter [2/3]	Information sent in either direction to indicate the invocation, acceptance or rejection of supplementary services, when no service associated parameter is to be sent.
Service Code	Information sent in the forward direction indicating a service code provided by the calling party.
Set [2/3]	The Set element is used to contain a set of information elements accompanying a component. It is required in the case of more than one information element being included in a component. The information elements themselves are defined in relevant supplementary service specifications.
Signaling Point Code [2/3]	Information sent in a release message to identify the signaling point in which the call failed.
Simple Segmentation Indicator [2]	Information sent in either direction to indicate that additional information will be forwarded in an information message (unsolicited).
Solicited Information Indicator	Information sent in an information message to indicate whether or not the message is a response to an information request message

Signaling Information	Definition
Special Processing Request [1]	Information sent in the forward direction from a private network access node to service node in the public switched network to indicate that the call requires special processing at that node e.g. private network number translation or verification of authorization codes.
Status	Information sent in a circuit group supervision message (e.g. Circuit Group Blocking) to indicate the specific circuits, within the range of circuits stated in the message, that are affected by the action specified in the message.
Sub-Group Identifier [1]	Information sent within a business group parameter to indicate the identity of the subgroup of which the user identified by the corresponding number is a member.
Suspend/Resume Indicator	Information sent in the Suspend and Resume Messages to indicate whether suspend/resume was initiated by an ISDN subscriber or by the network.
Suspend/Resume Indicators [1]	Information sent in either direction consisting of the suspend/resume indicator
Temporary Trunk Blocking After Release [2/3]	Information sent to the exchange at the other end of a circuit (trunk) to indicate low level of congestion at the sending exchange and that the circuit (trunk) should not be re-occupied by the receiving exchange for a short period of time after release.
Transaction Request [1]	Information sent in the Initial Address Message (lAM) to help continue call processing, using Transaction Capabilities (TC), associated with a given call during an assist or hand-off procedure.
Transfer Mode [1]	Information sent in the forward direction indicating circuit or packet transfer mode.
Transit at Intermediate Exchange Indicator [2]	Information sent to inform a transit node (Type B), whether it shall react on the rest of the instruction indicators or not, if the related message or parameter is unrecognized.

Signaling Information	Definition
Transit Network Selection [3]	Information sent in the forward direction indicating the transit network(s) requested for the routing of the call and consisting of the type of network identification, network identification plan, and network identification.
Transmission Medium Requirement [2]	Information sent in the forward direction indicating the type of transmission medium required for the connection (e.g. 64Kbps unrestricted speech).
Transmission Medium Requirement Prime [2]	Information sent in the forward direction indicating the fallback connection type in case of fallback.
Transmission Medium Used	Information sent in the backward direction indicating the resulting fallback connection type used for a call after the fallback has occurred.
Trunk Number [1]	Information used for circuit validation to identify the trunk number of the Common Language circuit identification.
Type Indicator [2]	Information sent to indicate the initiator for a circuit group supervision message, e.g., maintenance-oriented or hardware failure-oriented.
Type of Address [1]	Information that specifies the type of address digits contained in a Generic Address parameter.
Type of Digits [3]	Information that specifies the type of digits contained in a Generic Digits parameter.
Type of Network Identification [1]	Information sent in the forward direction indicating the format of the transit network identification.
Type of Network Identification [2/3]	Information sent to inform whether the identification of a network is by CCITT (ITU) standardization identification or by national network identification.
User Service Information	Information sent in the forward direction indicating the bearer capability requested by the calling party and including as a minimum, the coding standard and information transfer capability, transfer mode, and information transfer rate.
User Service Information Prime	Information sent in the forward direction indicating the preferred bearer capability requested by the calling party.

Signaling Information	Definition
User Teleservice Information [2]	Information sent in the initial address message indicating the Higher Layer Compatibility information requested by the calling party.
User-Network Interaction Indicator [1]	Information sent in the backward direction to indicate that the sending exchange is collecting additional information from the calling party before routing the call further.
User-to-User Indicators	Information sent in association with the response to a request for user-to-user signaling supplementary service(s).
User-to-User Information	Information generated by a user and transferred transparently through the intermediate exchanges between the originating and terminating local exchanges.

Message Parameters

Overview The ISUP parameters in this section are used in the *SS7 ISUP Message Format Configure* API message.

Definitions The table below indicates the IDs and descriptions for ISUP parameters. Some parameters are marked with a number indicating the telecommunications standard supported by a parameter:

[1]ANSI

[2]ITU

The parameters that are not marked by a number are supported in both ANSI and ITU standards.

Parameter Name	ID	Description
Access Delivery Information [2]	0x2E	Information sent in the backward direction indicating that a SETUP message was generated at the destination access.
Access Transport	0x03	Information generated on the access side of a call and transferred transparently in either direction between the originating and terminating local exchanges. The information is of significance to both users and local exchanges.
Automatic Congestion Level	0x27	Information optionally included in a Release Message and sent to the exchange at the other end of a circuit to indicate that a particular level of congestion exists at the sending exchange.
Backward Call Indicator (BCI)	0x11	Information sent in the backward direction consisting of the charge indicator, called party's status indicator, called party's category indicator, end-to-end method indicator, interworking indicator, end-to-end information indicator, ISDN User Part indicator, holding indicator, ISDN access indicator, echo control device indicator, and SCCP method indication.

Parameter Name	ID	Description
Business Group [1]	0xC6	Information sent in either direction to indicate the identity of the Multi-Location Business Group associated with a number (e.g. the Calling Party Number), the identity of a Subgroup within that Multilocation Business Group and the Line Privileges allocated to the number.
Call Diversion Information [2]	0x36	Information sent in the backward direction indicating the redirection reason and the notification subscription option of the redirecting user.
Call History Information [2]	0x2D	Information sent in the backward direction to indicate the accumulated propagation delay of a connection.
Call Reference	0x45	Circuit independent information identifying a particular call.
Called Party Number (CDPN)	0x04	Information sent in the forward direction to identify the called party and consisting of the odd/even indicator, nature of address indicator, numbering plan indicator, and address signals.
Calling Party Number (CGPN)	0x0A	Information sent in the forward direction to identify the calling party and consisting of the odd/even indictor, nature of address indicator, numbering plan indicator, address presentation restriction indicator, screening indicator, and address signals.
Calling Party's Category (CPC)	0x09	Information sent in the forward direction indicating the category of the calling party, e.g. ordinary subscriber, test call.
Cause Indicators	0x12	Information sent in either direction consisting of the coding standard, location, cause value and diagnostics. It indicates the reason for sending the message in which it is contained, e.g. the Release, Address complete or Confusion messages, and identifies the network in which the message originated, e.g. local network, transit network, remote local network.
Charge Number [1]	0xEB	Information sent in either direction indicating the chargeable number for the call and consisting of the odd/even indicator, nature of address indicator, numbering plan indicator, and address signals.

Parameter Name	ID	Description
Circuit Assignment Map [1]	0x25	Information identifying the map format and the circuits in a multi-rate connection. For the DS 1 map format it identifies the DS0 circuits constituting a NxDS0 connection. The circuit assignment map is used only for NxDS0 calls when non-contiguous time slot allocation is used.
Circuit Group Characteristics Indicator [1]	0xE5	Information sent in response to a request for circuit validation, indicating the characteristics of the concerned circuit group in terms of carrier type, double seizing control, alarm handling and continuity check requirements.
Circuit Group Message Supervision Message Type Indicator	0x15	Information sent in either direction in a circuit group supervision message and consisting of the circuit group blocking type indicator.
Circuit Identification Name [1]	0xE8	Information identifying the exchanges on which a circuit group terminates and the number of each trunk within that group.
Circuit State Indicator	0x26	Information indicating the state of a circuit according to the sending exchange.
Circuit Validation Response Indicator	0xE6	Information indicating the far-end results of a circuit validation test.
Closed User Group Interlock Code [2]	0x1A	Information uniquely identifying a closed user group within a network.
Common Language Location Information Code (CLLI) [1]	0xE9	Information used for circuit validation to identify a switching office by town, state, and building subdivision. (COMMON LANGUAGE is a registered trademark and CLLI is a trademark of Bell Communications Research, Inc.)
Connected Number [2]	0x21	Information sent in the backward direction to identify the connected party.
Connection Request	0x0D	Information sent in the forward direction on behalf of the Signaling Connection Control Part requesting the establishment of an end-to-end connection and consisting of the local reference, point code, protocol class, and credit.

Parameter Name	ID	Description
Continuity Indicators	0x10	Information sent in the forward direction indicating whether or not the continuity check on the outgoing circuit was successful. A continuity check successful indication also implies continuity of the preceding circuits and successful verification of the path across the exchange with the specified degree of reliability.
Echo Control Information [2]	0x37	Information sent in the forward direction indicating whether or not an outgoing half echo control device is included in the connection.
Egress [1]	0xC3	Information sent in the forward direction by the first incoming exchange in the terminating exchange area, to indicate network specific attributes associated with the terminating exchange, e.g. the interexchange carrier, the point of interconnection, and type of terminating access service.
End of Optional Parameters	0x00	The end of optional parameters field indicates that there are no more optional parameters in the message.
Event Information	0x24	Information sent in either direction consisting of the event indicator and event presentation restriction indicator.
Facility Indicators [2]	0x18	Information sent in facility related messages identifying the facility or facilities with which the message is concerned.
Forward Call Indicator	0x07	Information sent in the forward direction consisting of the incoming international call indicator, end-to-end method indicator, interworking indicator, end-to-end information indicator, ISDN User Part indicator, ISDN User Part preference indicator, ISDN access indicator, and SCCP method indicator.
Freephone Indicators [2]	0x41	No description available.
Generic Address [1]	0xC0	Information in the form of an address pertaining to a supplementary service (e.g., dialed number, destination number) and including type of address, nature of address and numbering plan indications.

Parameter Name	ID	Description
Generic Digits [1]	0xC1	Information in the form of digits pertaining to a supplementary service, (e.g. account code, authorization code, private network class mark) and including type of digits and encoding method indicators.
Generic Name [1]	0xC7	Information sent in the forward direction containing specific name related information.
Generic Notification [2]	0x2C	Information sent in either direction intended to provide supplementary service notification to a user.
Generic Number [2]	0xC0	A number information sent in either direction to enhance network operation or for supplementary services.
Generic Reference [2] (reserved)		For further study.
Hop Counter (ITU – reserved)	0x3D	Information sent in the forward direction to minimize the impact of lAM looping. The initial count determines the maximum number of contiguous SS7 interexchange circuits that are allowed to complete the call, assuming all subsequent intermediate exchanges decrement the hop counter.
Information Indicators	0x0F	Information sent in either direction consisting of the calling party address response indicator, connected address response indicator, calling party's category response indicator, charge information response indicator, and solicited information indicator.
Information Request Indicators	0x0E	Information sent in either direction consisting of the calling party address request indicator, connected address request indicator, calling party's category request indicator, charge information request indicator, malicious call identification request indicator, and holding indicator.
Jurisdiction [1]	0xC4	Information sent in the forward direction indicating the geographic origination of the call.
Location Number [2]	0x3F	Information sent to indicate the location of a user in the term of an E.164 number.
MCID Request Indicator [2]	0x3B	Information sent in the backward direction to request the identity of the calling party for the purpose of malicious call identification.

Parameter Name	ID	Description
MCID Response Indicator [2]	0x3C	Information sent in the forward direction to respond to a MCID request and indicating whether or not the MCID information is available.
Message Compatibility Information [2]	0x38	Information sent in either direction indicating how an exchange should react in case this message is unrecognized.
MLPP Precedence [2]	0x3A	Information sent in the precedence parameter identifying the network resources to which the Multi-level Precedence and Preemption supplementary service is applicable for the call.
Nature of Connection Indicator	0x06	Information sent in the forward direction consisting of the satellite indicator, **continuity check indicator**, and echo control device indicator.
Network Specific Facilities [2]	0x2F	Service related information transparently transferred in either direction between the local exchange and the identified network, which contracts the service. The information is significant to both user and the identified network.
Network Transport [1]	0xEF	A Parameter sent in either direction for the purpose of transporting other ISDN User Part parameters transparently across transit exchanges.
Notification Indicator [1]	0xE1	Information Sent in either direction intended to provide supplementary service related notification to a user.
Operator Services Information [1]	0xC2	Information sent in the forward direction between operator services entities primarily identifying charging and service type options.
Optional Backward Call Indicators	0x29	Information sent in the backward direction consisting of the in-band information indicator, the call forwarding may occur indicator and the user-network interaction indicator.
Optional Forward Call Indicators [2]	0x08	Information sent in the forward direction consisting of CUG, segmentation, and connected line identity request information.
Original Called Number	0x28	Information sent in the forward direction to indicate, in the case of call redirection (e.g., call forwarding), the number of the user who initiated the initial redirection.

Parameter Name	ID	Description
Originating Line Information (OLI) [1]	0xEA	Information sent in the forward direction, indicating a toll class of service for the call.
Origination ISC Point Code [2]	0x2B	Information sent in the initial address message of an international call, indicating the point code of the originating ISC.
Outgoing Trunk Group Number (OTGN)		Information sent in the backward direction indicating the trunk group selected at an outgoing gateway. For intra-network use only.
Parameter Compatibility Information [2]	0x39	Information sent in either direction indicating how an exchange should react in case the parameter is unrecognized.
Precedence [1]	0x3A	Information sent in the forward direction in association with the invocation of the Multi-Level Precedence and Preemption (MIYP) supplementary service, consisting of the look-ahead for busy indication, the precedence level, network identity, and the MI2P service domain.
Propagation Delay Counter [2]	0x31	Information sent in the forward direction to indicate the propagation delay of a connection. This information is accumulated while the parameter is transferred through the network. The propagation delay information is represented by a counter counting in integer multiples of 1 ms.
Range and Status (R&S)	0x16	Information sent in either direction in circuit group supervision messages consisting of the range and status.
Redirecting Number	0x0B	Information sent in the forward direction indicating the number from which the call was last redirected and consisting of the nature of address indicator, numbering plan indicator, address presentation restriction indicator, and address information (signals).
Redirection Information	0x13	Information sent in the forward direction consisting of the original redirecting reason redirection counter and redirecting reason.
Redirection Number [2]	0x0C	Information sent in the backward direction indicating the number towards which the call must be rerouted or has been forwarded.

Parameter Name	ID	Description
Redirection Number Restriction [2]	0x40	Information sent in the backward direction indicating whether the diverted-to user allows the presentation of his number.
Remote Operations	0x32	The remote operations parameter is used to indicate the invocation of a supplementary service identified by an operation value and also carry the result or error indications depending on the outcome of the operation.
Service Activation	0x33	Information sent in either direction to indicate the invocation, acceptance or rejection of supplementary services, when no service associated parameter is to be sent.
Signaling Point Code [2]	0x1E	Information sent in a release message to identify the signaling point in which the call failed.
Special Processing Request [1]	0xE5	Information sent in the forward direction from a private network access node to service node in the public switched network to indicate that the call requires special processing at that node e.g. private network number translation or verification of authorization codes.
Subsequent Number [2]	0x05	Information sent which includes E/O indicator, address signals and filler.
Suspend/Resume Indicators	0x22	Information sent in either direction consisting of the suspend/resume indicator
Transaction Request [1]	0xE3	Information sent in the Initial Address Message (lAM) to help continue call processing, using Transaction Capabilities (TC), associated with a given call during an assist or hand-off procedure.
Transit Network Selection (TNS)	0x23	Information sent in the forward direction indicating the transit network(s) requested for the routing of the call and consisting of the type of network identification, network identification plan, and network identification.
Transmission Medium Requirement [2]	0x02	Information sent in the forward direction indicating the type of transmission medium required for the connection (e.g. 64Kbps unrestricted speech).
Transmission Medium Requirement Prime [2]	0x3E	Information sent in the forward direction indicating the fallback connection type in case of fallback.

Parameter Name	ID	Description
Transmission Medium Used	0x35	Information sent in the backward direction indicating the resulting fallback connection type used for a call after the fallback has occurred.
User Service Information (USI)	0x75	Information sent in the forward direction indicating the bearer capability requested by the calling party and including as a minimum the coding standard, information transfer capability, transfer mode, and information transfer rate.
User Service Information Prime	0x30	Information sent in the forward direction indicating the preferred bearer capability requested by the calling party.
User Teleservice Information [2]	0x34	Information sent in the initial address message indicating the Higher Layer Compatibility information requested by the calling party.
User-to-User Indicators	0x2A	Information sent in association with the response to a request for user-to-user signaling supplementary service(s).
User-to-User Information	0x20	Information generated by a user and transferred transparently through the intermediate exchanges between the originating and terminating local exchanges.

Cause Codes

Overview

The following cause values, names and definitions are identified in Q.850 as pertaining to ISUP.

ITU Standard Causes

The following cause values, names and definitions are identified in Q.850 as pertaining to ISUP.

NORMAL CLASS

Cause 1 Unallocated (unassigned) number - This cause indicates that the called party cannot be reached because, although the called party number is in a valid format, it is not currently allocated (assigned).

Cause 2 No route to specified transit network - This cause indicates that the equipment sending this cause has received a request to route the call through a particular transit network which it does not recognize. The equipment sending this cause does not recognize the transit network either because the transit network does not exist or because that particular transit network, while it does exist, does not serve the equipment that is sending this cause. This cause is supported on a network dependent basis.

Cause 3 No route to destination - This cause indicates that the called party cannot be reached because the network through which the call has been routed does not serve the destination desired. This cause is supported on a network-dependent basis.

Cause 4 Send special information tone - This cause indicates that the called party cannot be reached for reasons that are of long term nature and that the special information tone should be returned to the calling party.

Cause 5 Misdialed trunk prefix - No procedures specified for US networks.

Cause 8 Preemption - This cause indicates that the call is being preempted.

Cause 9 Preemption - Circuit reserved for reuse. This cause indicates that the call is being preempted and the circuit is reserved for reuse by the preempting exchange.

Cause 16 Normal call clearing - This cause indicates that the call is being cleared because one of the users involved in the call has requested that the call be cleared. Under normal situations, the source of this cause is not the network.

Cause 17 User busy - This cause is used to indicate that the called party is unable to accept another call because the user busy condition has been encountered. This cause value may be generated by the called user or by the network. In the case of user determined user busy, it is noted that the user equipment is compatible with the call.

Cause 18 No user responding - This cause is used when a called party does not respond to a call establishment message with either an alerting or connect indication within the prescribed period of time allocated.

Cause 19 No answer from user (user alerted) - This value is used when the called party has been alerted but does not respond with a connect indication within a prescribed period of time. Note - This cause is not necessarily generated by Q.93l procedures but may be generated by internal network timers.

Cause 20 Subscriber absent - This cause value is used when a mobile station has logged off, radio contact is not obtained with a mobile station or if a personal telecommunication user is temporarily not addressable at any user-network interface.

Cause 21 Call rejected - This cause indicates that the equipment sending this cause does not wish to accept this call, although it could have accepted the call because the equipment sending this cause is neither busy nor incompatible. The network may also generate this cause, indicating that the call was cleared due to a supplementary service constraint. The diagnostic field may contain additional information about the supplementary service and reason for rejection.

Cause 22 Number changed - This cause is returned to a calling party when the called party number indicated by the calling party is no longer assigned. The new called party number may optionally be included in the diagnostic field. If a network does not support this cause value, cause number 1, unallocated (unassigned) number shall be used.

Cause 27 Destination out of order - This cause indicates that the destination indicated by the user can not be reached because the interface to the destination is not functioning correctly. The term “not

functioning correctly" indicates that a signaling message was unable to be delivered to the remote party; e.g., a physical layer or data link layer failure at the remote party, or user equipment off-line.

Cause 28 Invalid number format (address incomplete) - This cause indicates that the called party cannot be reached because the called party number is not in a valid format or is not complete.

This condition may be determined:

- Immediately after reception of an ST signal, or
- On time-out after the last received digit.

Cause 29 Facility rejected - This cause is returned when the network cannot provide a supplementary service requested by the user.

Cause 31 Normal, unspecified - This cause is used to report a normal event only when no other cause in the normal class applies.

RESOURCE UNAVAILABLE CLASS

Cause 34 No circuit/channel available - This cause indicates that there is no appropriate circuit/channel presently available to handle the call.

Cause 38 Network Out of order - This cause indicates that the network is not functioning correctly and that the condition is likely to last a relatively long period of time, e.g., immediately re-attempting the call is not likely to be successful.

Cause 41 Temporary failure - This cause indicates that the network is not functioning correctly and that the condition is not likely to last a long period of time, e.g., the user may wish to try another call attempt almost immediately.

Cause 42 Switching equipment congestion - This cause indicates that the switching equipment generating this cause is experiencing a period of high traffic.

Cause 43 Access information discarded - This cause indicates that the network could not deliver access information to the remote user as requested, i.e., user-to-user information, low layer compatibility, high layer compatibility, or sub-address, as indicated in the diagnostic. It is noted that the particular type of access information discarded is optionally included in the diagnostic.

Cause 44 Requested circuit/channel not available - This cause is returned when the circuit or channel indicated by the requesting entity cannot be provided by the other side of the interface.

Cause 46 Precedence call blocked - This cause indicates that there are no preemptable circuits or that the called user is busy with a call of equal or higher preemptable level.

Cause 47 Resource unavailable, unspecified - This cause is used to report a resource unavailable event only when no other cause in the resource unavailable class applies.

SERVICE OR OPTION UNAVAILABLE CLASS

Cause 50 Requested facility not subscribed - This cause indicates that the user has requested a supplementary service that is implemented by the equipment that generated this cause, but the user is not authorized to use.

Cause 53 Outgoing calls barred within CUG – No procedure specified for US networks.

Cause 55 Incoming calls barred within CUG –

No procedure specified for US networks.

Cause 57 Bearer capability not authorized -

This cause indicates that the user has requested a bearer capability that is implemented by the equipment that generated this cause, but the user is not authorized to use.

Cause 58 Bearer capability not presently available -

This cause indicates that the user has requested a bearer capability that is implemented by the equipment which generated this cause but which is not available at this time.

Cause 62 inconsistency in designated outgoing access information and subscriber class -

This cause indicates that there is an inconsistency in the designated outgoing access information and subscriber class.

Cause 63 Service or option not available, unspecified -

This cause is used to report a service or option not available event only when no other cause in the service or option not available class applies.

SERVICE/OPTION NOT IMPLEMENTED CLASS

Cause 65 Bearer capability not implemented - This cause indicates that the equipment sending this cause does not support the bearer capability requested.

Cause 69 Requested facility not implemented - This cause indicates that the equipment sending this cause does not support the requested supplementary service.

Cause 70 Only restricted digital information bearer capability is available - This cause indicates that the calling party has requested an unrestricted bearer service but that the equipment sending this cause only supports the restricted version of the requested bearer capability.

Cause 79 Service or option not implemented, unspecified - This cause is used to report a service or option not implemented event only when no other cause in the service or option not implemented class applies.

INVALID MESSAGE (E.G. PARAMETER OUT OF RANGE) CLASS

Cause 87 User not member of CUG – No procedures specified for US networks.

Cause 88 Incompatible destination - This cause indicates that the equipment sending this cause has received a request to establish a call which has low layer compatibility, high layer compatibility, attributes (e.g., data rate) which can not be accommodated.

Cause 90 Non-existent CUG – No procedures specified for US networks.

Cause 91 Invalid transit network selection - This cause indicates that a transit network identification was received which is of an incorrect format as defined in Annex C of TI.607

Cause 95 Invalid message, unspecified - This cause is used to report an invalid message event only when no other cause in the invalid message class applies.

PROTOCOL ERROR (E.G. UNKNOWN MESSAGE) CLASS

Cause 97 Message type non-existent or not implemented -

This cause indicates that the equipment sending this cause has received a message with a message type it does not recognize either because this is a message not defined or defined but not implemented by the equipment sending this cause.

Cause 99 Information Element/Parameter non-existent or not implemented -

This cause indicates that the equipment sending this cause has received a message which includes information element(s)/parameter(s) not recognized because the information element identifier(s)/parameter name(s) are not defined or are defined but not implemented by the equipment sending the cause. This cause indicates that the information element(s)/parameter(s) were discarded. However, the information element is not required to be present in the message in order for the equipment sending the cause to process the message.

Cause 102 Recovery on timer expiry -

This cause indicates that a procedure has been initiated by the expiry of a timer in association with error handling procedures.

Cause 103 Parameter non-existent or not implemented – pass on –

No procedures specified for US networks.

Cause 110 Message with unrecognized parameter discarded -

This cause indicates that the equipment sending this cause has discarded a received message which includes a parameter that is not recognized.

Cause 111 Protocol error unspecified -

This cause is used to report a protocol error event only when no other cause in the protocol error class applies.

INTERWORKING CLASS

Cause 127 Interworking, unspecified -This cause indicates that there has been interworking with a network which does not provide causes for actions it takes. Thus, the precise cause for a message that is being sent cannot be ascertained.

ANSI Standard Causes

The following cause values, names and definitions are identified in ____ as pertaining to ISUP.

NORMAL CLASS

Cause 23 Unallocated destination number - This cause indicates failure of a business group call because the business group identifier is unknown.

Cause 24 Undefined business group - This cause indicates failure of a business group call because the destination number is unallocated in the business group numbering plan.

Cause 25 Exchange routing error - This cause indicates that the called party cannot be reached because of an error in exchange number translation tables used to route the call (results when the hop counter parameter value is decremented to 0).

RESOURCE UNAVAILABLE CLASS

Cause 45 Preemption - (Used by T1.113-1988 and Tl.619-1992.) This cause indicates that the equipment sending this cause has preempted the circuit for a new call and the existing call should be cleared.

Cause 46 Precedence call blocked - (Used by T1.113-1988 and T1.169-1992.) In case of a call with a precedence level higher than the lowest level (ROUTINE), this cause indicates that all circuits are busy on calls of equal or higher priority.

SERVICE OR OPTION NOT AVAILABLE CLASS

Cause 51 Call type incompatible with service request - This cause indicates that an action was rejected because it was not compatible with the bearer capability of the call.

Cause 54 Call blocked due to group restrictions -

This cause indicates failure of a business group call because it did not pass line privileges and/or subgroup screening.

V5.2 Protocol Implementation Compliance Statement (PICS)

The table below indicates features that are and are not supported.

Main Features Description	Supported
ISDN BA Ports	
ISDN PRA Ports	
PSTN Ports	Y
Bearer channel connection	Y
Semi-permanent leased lines	
Pre-assigned bearer channel	
Communication path definition	Y
C-path(s) for p-type data	
C-path(s) for f-type data	
C-path(s) for Ds-type data	
C-path for PSTN signaling	Y
C-path for control	Y
C-path for bearer channel connection	Y
C-path for protection	Y
C-path for link control	Y
Logical Communication channel provisioning	Y
Default active communication channel allocation by provisioning	Y
Default active communication channel	Y
Default standby Communication channel	Y

Number of physical communication channels up to 3 times the number of 2048 kbit/s links	
Number of backup communication channel up to 3	
Protection Switching of communication channels	Y
Envelope function	Y
ISDN ports only partially provisioned for on demand service (PL service)	
ISDN PRA ports only partially provisioned for on-demand service (PL service)	
Multi-slot connection	
Multi-link V5.2 Interface	Y

Layer 1	
Layer 1 Balanced	Y
Layer 1 coaxial	Y
Layer 1 link maintenance requirements	Y
Detection of loss of signals;1 ms below 20 dB	Y
Detection of loss of signals; 10 consecutive zeros	
Link control requirements and procedures	Y
Layer 2	
Frame structure for peer to peer communication	Y
Format of fields for data link envelop	Y
Envelope address value for control protocol	Y
Envelope address value for PSTN protocol	Y
Envelope address values for ISDN ports	
Envelope address value BCC protocol	Y
Envelope address value for Protection protocol	Y
Envelope address values for link control protocol	Y
Data link sub layer of LAPV5 for control protocol	Y
Data link sub layer of LAPV5 for PSTN protocol	Y
Data link sub layer of LAPV5 for bearer connection control protocol	Y
Frame relay function in the AN	
Data link sub layer of LAPV5 for protection protocol	Y
Data link sub layer of LAPV5 for link control protocol	Y

Layer 3	
PSTN Protocol	
Control of time critical sequences by AN	Y
PSTN protocol entity	Y
PSTN call control entity	Y
Control Protocol	
Control protocol entity	Y
Port Control Protocol	
ISDN BA user port status indication and control	
ISDN PRA user port status indication and control	
Performance monitoring for ISDN BA user ports	
Performance monitoring for ISDN BA user ports	
Performance monitoring for ISDN PRA user port	
PSTN user port status indication and control	Y
Common Control Protocol	
Variant and interface ID control	Y
Verify re-provisioning	
Re-provisioning synchronization	
BCC Protocol	
Bearer Channel Connection	Y
Bearer Channel Connection Auditing	Y
Protection Protocol	
Protection switching of group 1	Y
Protection switching of group 2	

Link Control Protocol	
Link control protocol	Y
PSTN Protocol	
Messages	
ESTABLISH	Y
ESTABLISH Acknowledge	Y
SIGNAL	Y
SIGNAL Acknowledge	Y
STATUS	Y
STATUS ENQUIRY	Y
DISCONNECT	Y
DISCONNECT COMPLETE	Y
PROTOCOL PARAMETER	
PSTN General	
Protocol discriminator	Y
Layer 3 address	Y
Pulse notification	Y
Line information	Y
State	Y
Autonomous signaling sequence	Y
Sequence response	Y
Sequence-number	Y
Cadenced ringing	Y
Pulsed signal	Y

Steady signal	Y
Digit signal	Y
Recognition time	Y
Enable autonomous Acknowledge	Y
Disable autonomous Acknowledge	Y
Cause	Y
Resource unavailable	Y
Information Elements: Pulse Types	
Pulsed normal polarity	Y
Pulsed reversed polarity	Y
Pulsed battery on c-wire	
Pulsed on hook	Y
Pulsed reduced battery	
Pulsed no battery	
Initial ring	
Meter pulse	
50 Hz pulse	
Register recall	Y
Pulsed off hook	Y
Pulsed B-wire connected to earth	
Earth loop pulse	
Pulsed B-wire connected to battery	
Pulsed A-wire connected to earth	
Pulsed A-wire connected to battery	

Pulsed C-wire connected to earth	
Pulsed C-wire disconnected	
Pulsed normal battery	
Pulsed A-wire disconnected	
Pulsed normal battery	
Pulsed A-wire disconnected	
Pulsed B-wire disconnected	
Information Elements: Steady Signals	
Normal polarity	Y
Reversed polarity	Y
Battery on C-wire	
No battery on C-wire	
Off hook	Y
On hook	Y
Battery on A-wire	
A-wire on earth	
No battery on A-wire	
No battery on B-wire	
Reduced battery	
No battery	
Alternate reduced poser/no power	
Normal battery	
Stop ringing	Y
Start pilot frequency	

Stop pilot frequency	
Low impedance on B-wire	
B-wire connected to earth	
B-wire disconnected from earth	
Normal battery on B-wire	
Low loop impedance	
High loop impedance	
Anomalous loop impedance	
A-wire disconnected from earth	
C-wire on earth	
C-wire disconnected from earth	
Information Elements: Cause Types	
Response to status enquiry	Y
Protocol discriminator error	Y
L3 address error	Y
Message type unrecognized	Y
Out of sequence information element	Y
Repeated optional information element	Y
Mandatory information element missing	Y
Unrecognized information element	Y
Mandatory information element content error	Y
Optional information element content error	Y
Message not compatible with state	Y
Repeated mandatory information element	Y

Too many information elements	Y
Information Elements: Information Element Fields	
Suppression indicator	Y
Acknowledge request indicator	Y
Digit Acknowledge request indicator	Y

Control Protocol	
Messages	
Common control and port control messages	Y
Information Elements; General	
Protocol discriminator	Y
Layer 3 addresses	Y
Information Elements; Port Control	
FE101 activate access	
FE102 activation initiated by user	
FE103 DS activated	
FE104 access activated	
FE105 deactivate access	
FE105 access deactivated	
FE201/202 unblock	Y
FE203/204 block	Y
FE205 block request	Y
FE205 performance grading	
FE207 D channel block	

FE208 D channel unblock	
FE209 TE out of service	
FE210 Failure inside network	
Information elements; Common Control	
Verify re-provisioning	
Ready for re-provisioning	
Not ready for re-provisioning	
Switch-over to new variant	
Re-provisioning started	
Cannot re-provision	
Request variant and interface ID	Y
Variant and interface ID	Y
Blocking started	
Restart	Y
Restart Acknowledge	Y
BCC Protocol	
Messages	
BCC protocol messages	Y
Information Elements	
BCC reference number	Y
Message type	Y
User port identification	Y
ISDN port time slot identification	
V5 Timeslot identification	Y

Multi slot map	
Reject cause	Y
Protocol error cause	Y
Connection incomplete	Y
Protection Protocol	
Messages	
Protection switching protocol messages	Y
Information Elements	
Protection switching protocol information elements	Y
Link Control Protocol	
Messages	
Link control protocol messages	Y
Information Elements	
Link control protocol information elements.	Y

V5.2 Example Configuration Trace

Important! This release of V5.2 supports LE only.

Important! AN is not supported for V5.2. Any text or illustrations (for example, call flows) indicating the AN side is for clarification purposes only.

Overview

The following trace shows an example configuration for a system using the V5.2 protocol. In this example, both ends of the V5.2 interface are shown in one CSP system; the LE side and the AN side.

The following is a list of objects that the host must manage for this implementation of the V5.2 protocol:

V5 IDs

This example shows the host creating four new V5 IDs within the system. V5 IDs 0 and 2 are assigned for the LE side, and V5 IDs 1 and 3 are assigned for the AN side. These V5 IDs are logical entities that the host uses to manage the V5 interfaces within the system. Once the V5 IDs are created within the system, the host must associate link IDs with each V5 ID.

Attributes

- Create
- Destroy
- Bring In Service
- Take Out of Service
- Modify Variant ID
- Modify Variant Type
- Modify Network Side
- Modify V5 Interface ID

Contains

- Associated Links with C Channels
- Links without C Channels
- User Ports

Links

Each V5 ID has associated links that carry the voice data. These links are logically represented as logical link IDs for each V5 ID. Each link ID corresponds to a unique logical span ID. Links can carry data only (no C Channels residing on them), or may contain C Channels that carry the signaling information for the V5.2 protocol and data.

Attributes

- Add links
- Remove links

Contains

- Associated C Channels

C Channels

Each V5 ID within the system has a corresponding C Channel. The host must manage logical C Channel IDs for each V5 ID within the system. For the V5.2 protocol, there is a primary C Channel and a secondary C Channel for redundancy. The C Channels must reside on timeslot 16 of the links on which they are assigned.

Attributes

- Add C Channel
- Remove C Channel

Logical Span IDs

The host must manage logical span IDs within the CSP system. Logical span IDs are assigned to the physical trunk interfaces to terminate E1 spans. These logical span IDs are associated with the V5 link IDs. The host associates the span IDs with the link IDs when configuring the links using the add link TLV contained within the V5 configure message.

Attributes

- Assign
- De-assign
- Bring In Service
- Bring Out of Service
- Modify Span Format

User Ports

A group of user port IDs are assigned by the host, to each V5 ID within the system. A user port ID represents an actual user of the service.

Attributes

- Add User Port Range Type
- Remove User Port Range Type

This example shows:

- De-assigning any previously configured spans within the system
- Assigning eight logical span IDs to the physical E1 span IDs (0-7)
- Formatting the E1 spans to clear channel
- Creating four V5 IDs (0-3)
- Configuring each V5 ID:

V5 ID 0

- V5 Interface ID: 000001 (Actual ID used across the network)
- V5 Variant ID: 0001
- V5 Variant Type:02 (V5.2)
- V5 Network Side:00 (LE)

V5 ID 1

- V5 Interface ID: 000001 (Actual ID used across the network)
- V5 Variant ID: 0001
- V5 Variant Type: 02 (V5.2)
- V5 Network Side:01 (AN)

V5 ID 2

- V5 Interface ID: 000002 (Actual ID used across the network)
- V5 Variant ID: 0002
- V5 Variant Type:02 (V5.2)
- V5 Network Side:00 (LE)

V5 ID 3

- V5 Interface ID: 000002 (Actual ID used across the network)
- V5 Variant ID:0002
- V5 Variant Type:02 (V5.2)
- V5 Network Side:01 (AN)

Adding User Port Ranges to each V5 ID

V5 ID 0,1,2,3

- User Port Range: 100 user port IDs 0x00-0x63

Adding Links to each V5 ID

V5 ID 0

- Link ID 1: Logical Span ID 0
- Link ID 2: Logical Span ID 2

V5 ID 1

- Link ID 1: Logical Span ID 1
- Link ID 2: Logical Span ID 3

V5 ID 2

- Link ID 1: Logical Span ID 4
- Link ID 2: Logical Span ID 6

V5 ID 3

- Link ID 1: Logical Span ID 5
- Link ID 2: Logical Span ID 7

Adding C Channels to each V5 ID

V5 ID 0

- C Channel Type: 1 (Primary C Channel)
- Logical Span ID for Primary Link: 0
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 0
- Link ID 1

- C Channel Type: 2 (Secondary C Channel)
- Logical Span ID for Primary Link: 2
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 1
- Link ID: 2

V5 ID 1

- C Channel Type: 1 (Primary C Channel)
- Logical Span ID for Primary Link: 1
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 0
- Link ID: 1

- C Channel Type: 2 (Secondary C Channel)
- Logical Span ID for Primary Link: 3
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 1
- Link ID: 2

V5 ID 2

- C Channel Type: 1 (Primary C Channel)
- Logical Span ID for Primary Link: 4
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 0
- Link ID: 1

- C Channel Type: 2 (Secondary C Channel)
- Logical Span ID for Primary Link: 6
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 1
- Link ID: 2

V5 ID 3:

- C Channel Type: 1 (Primary C Channel)
- Logical Span ID for Primary Link: 5
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 0
- Link ID: 1

- C Channel Type: 2 (Secondary C Channel))
- Logical Span ID for Primary Link: 7
- Primary Physical Channel: 0x1E (Timeslot 16)
- C Channel ID: 1
- Link ID: 2

Bringing Logical Span IDs (0-7) and V5 IDs (0-3) In Service

Important! **H->X** This symbol represents a host-initiated message (host-to-switch)

Important! **X->H** This symbol represents a switch-initiated message (switch-to-host)

```
De-assign All Logical Span IDs:
H->X [ 00 0d 00 a8 00 00 ff 00 01 11 04 ff ff ff ff ]
X->H [ 00 0f 00 a8 00 00 ff 00 10 00 01 11 04 ff ff ff ff ]


Assign Logical Span IDs 0-7:
H->X [ 00 0d 00 a8 00 01 ff 00 01 11 04 00 00 00 00 ]
X->H [ 00 0f 00 a8 00 01 ff 00 10 00 01 11 04 00 00 00 00 ]

H->X [ 00 0d 00 a8 00 02 ff 00 01 11 04 00 01 00 01 ]
X->H [ 00 0f 00 a8 00 02 ff 00 10 00 01 11 04 00 01 00 01 ]

H->X [ 00 0d 00 a8 00 03 ff 00 01 11 04 00 02 00 02 ]
X->H [ 00 0f 00 a8 00 03 ff 00 10 00 01 11 04 00 02 00 02 ]

H->X [ 00 0d 00 a8 00 04 ff 00 01 11 04 00 03 00 03 ]
X->H [ 00 0f 00 a8 00 04 ff 00 10 00 01 11 04 00 03 00 03 ]

H->X [ 00 0d 00 a8 00 05 ff 00 01 11 04 00 04 00 04 ]
X->H [ 00 0f 00 a8 00 05 ff 00 10 00 01 11 04 00 04 00 04 ]

H->X [ 00 0d 00 a8 00 06 ff 00 01 11 04 00 05 00 05 ]
X->H [ 00 0f 00 a8 00 06 ff 00 10 00 01 11 04 00 05 00 05 ]

H->X [ 00 0d 00 a8 00 07 ff 00 01 11 04 00 06 00 06 ]
X->H [ 00 0f 00 a8 00 07 ff 00 10 00 01 11 04 00 06 00 06 ]

H->X [ 00 0d 00 a8 00 08 ff 00 01 11 04 00 07 00 07 ]
X->H [ 00 0f 00 a8 00 08 ff 00 10 00 01 11 04 00 07 00 07 ]
```

```
Format E1 Spans IDs 0-7 to Clear Channel:
H->X [ 00 0f 00 d8 00 09 ff 00 01 0c 02 00 00 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 09 ff 00 10 ]

H->X [ 00 0f 00 d8 00 0a ff 00 01 0c 02 00 01 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0a ff 00 10 ]

H->X [ 00 0f 00 d8 00 0b ff 00 01 0c 02 00 02 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0b ff 00 10 ]

H->X [ 00 0f 00 d8 00 0c ff 00 01 0c 02 00 03 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0c ff 00 10 ]

H->X [ 00 0f 00 d8 00 0d ff 00 01 0c 02 00 04 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0d ff 00 10 ]

H->X [ 00 0f 00 d8 00 0e ff 00 01 0c 02 00 05 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0e ff 00 10 ]

H->X [ 00 0f 00 d8 00 0f ff 00 01 0c 02 00 06 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 0f ff 00 10 ]

H->X [ 00 0f 00 d8 00 10 ff 00 01 0c 02 00 07 09 00 fc d0 ]
X->H [ 00 07 00 d8 00 10 ff 00 10 ]


V5 Configure (Create V5 IDs 0000, 0001, 0002, 0003):
H->X [ 00 12 00 7b 00 11 ff 00 01 01 01 02 01 01 00 01 00 02 00 00 ]
X->H [ 00 07 00 7b 00 11 ff 00 10 ]

H->X [ 00 12 00 7b 00 12 ff 00 01 01 01 02 01 01 00 01 00 02 00 02 ]
X->H [ 00 07 00 7b 00 12 ff 00 10 ]

H->X [ 00 12 00 7b 00 13 ff 00 01 01 01 03 01 01 00 01 00 02 00 01 ]
X->H [ 00 07 00 7b 00 13 ff 00 10 ]

H->X [ 00 12 00 7b 00 14 ff 00 01 01 01 03 01 01 00 01 00 02 00 03 ]
X->H [ 00 07 00 7b 00 14 ff 00 10 ]
```

```
V5 Configure (Configure V5 ID 0000, 0002 are LE SIDE) with 4 TLV s:
V5Interface ID, V5 Variant ID, V5 Variant Type, V5 Network Side (LE)
H->X [ 00 23 00 7b 00 15 ff 00 01 2c 02 00 00 01 04 00 10 00 03 00 00 01
00 03 00 01 01 00 04 00 01 02 00 05 00 01 00 ]
X->H [ 00 07 00 7b 00 15 ff 00 10 ]

H->X [ 00 23 00 7b 00 16 ff 00 01 2c 02 00 02 01 04 00 10 00 03 00 00 02
00 03 00 01 02 00 04 00 01 02 00 05 00 01 00 ]
X->H [ 00 07 00 7b 00 16 ff 00 10 ]


V5 Configure (Configure V5 ID 0001, 0003 are AN SIDE) with 4 TLVs:
V5 Interface ID, V5 Variant ID, V5 Variant Type, V5 Network Side (AN)
H->X [ 00 23 00 7b 00 17 ff 00 01 2c 02 00 01 01 04 00 10 00 03 00 00 01
00 03 00 01 01 00 04 00 01 02 00 05 00 01 01 ]
X->H [ 00 07 00 7b 00 17 ff 00 10 ]

H->X [ 00 23 00 7b 00 18 ff 00 01 2c 02 00 03 01 04 00 10 00 03 00 00 02
00 03 00 01 02 00 04 00 01 02 00 05 00 01 01 ]
X->H [ 00 07 00 7b 00 18 ff 00 10 ]
```

```
V5 Configure (Add USER PORTS TO V5 IDs 0000, 0001, 0002, 0003):
H->X [ 00 17 00 7b 00 19 ff 00 01 2c 02 00 00 01 01 00 06 00 06 01 01 00 00 00 63 ]
X->H [ 00 07 00 7b 00 19 ff 00 10 ]

H->X [ 00 17 00 7b 00 1a ff 00 01 2c 02 00 01 01 01 00 06 00 06 01 01 00 00 00 63 ]
X->H [ 00 07 00 7b 00 1a ff 00 10 ]

H->X [ 00 17 00 7b 00 1b ff 00 01 2c 02 00 02 01 01 00 06 00 06 01 01 00 00 00 63 ]
X->H [ 00 07 00 7b 00 1b ff 00 10 ]

H->X [ 00 17 00 7b 00 1c ff 00 01 2c 02 00 03 01 01 00 06 00 06 01 01 00 00 00 63 ]
X->H [ 00 07 00 7b 00 1c ff 00 10 ]


V5 Configure ( Add Links to V5 IDs 0000, 0001, 0002, 0003):
H->X [ 00 1d 00 7b 00 1d ff 00 01 2c 02 00 00 01 02 00 0c 00 04 00 00 01 01 00 0c
00 04 00 02 02 02]
X->H [ 00 07 00 7b 00 1d ff 00 10 ]

H->X [ 00 1d 00 7b 00 1e ff 00 01 2c 02 00 01 01 02 00 0c 00 04 00 01 01 01 00 0c
00 04 00 03 02 02]
X->H [ 00 07 00 7b 00 1e ff 00 10 ]

H->X [ 00 1d 00 7b 00 1f ff 00 01 2c 02 00 02 01 02 00 0c 00 04 00 04 01 01 00 0c
00 04 00 06 02 02]
X->H [ 00 07 00 7b 00 1f ff 00 10 ]

H->X [ 00 1d 00 7b 00 20 ff 00 01 2c 02 00 03 01 02 00 0c 00 04 00 05 01 01 00 0c
00 04 00 07 02 02]
X->H [ 00 07 00 7b 00 20 ff 00 10 ]
```

```
V5 Configure ( Add C Channels to V5 IDs 0000, 0001, 0002, 0003):
H->X [ 00 21 00 7b 00 21 ff 00 01 2c 02 00 00 01 02 00 0a 00 06 01 00 00 1e
00 01 00 0a 00 06 02 00 02 1e 01 02 ]
X->H [ 00 07 00 7b 00 21 ff 00 10 ]

H->X [ 00 21 00 7b 00 22 ff 00 01 2c 02 00 01 01 02 00 0a 00 06 01 00 01 1e
00 01 00 0a 00 06 02 00 03 1e 01 02 ]
X->H [ 00 07 00 7b 00 22 ff 00 10 ]

H->X [ 00 21 00 7b 00 23 ff 00 01 2c 02 00 02 01 02 00 0a 00 06 01 00 04 1e
00 01 00 0a 00 06 02 00 06 1e 01 02 ]
X->H [ 00 07 00 7b 00 23 ff 00 10 ]

H->X [ 00 21 00 7b 00 24 ff 00 01 2c 02 00 03 01 02 00 0a 00 06 01 00 05 1e
00 01 00 0a 00 06 02 00 07 1e 01 02 ]
X->H [ 00 07 00 7b 00 24 ff 00 10 ]
```

```
Service State Configure (Bring spans 0-7 IN SERVICE):
H->X [ 00 0d 00 0a 00 25 ff 00 01 0c 02 00 00 f0 00 ]
X->H [ 00 07 00 0a 00 25 ff 00 10 ]

H->X [ 00 0d 00 0a 00 26 ff 00 01 0c 02 00 01 f0 00 ]
X->H [ 00 07 00 0a 00 26 ff 00 10 ]

H->X [ 00 0d 00 0a 00 27 ff 00 01 0c 02 00 02 f0 00 ]
X->H [ 00 07 00 0a 00 27 ff 00 10 ]

H->X [ 00 0d 00 0a 00 28 ff 00 01 0c 02 00 03 f0 00 ]
X->H [ 00 07 00 0a 00 28 ff 00 10 ]

H->X [ 00 0d 00 0a 00 29 ff 00 01 0c 02 00 04 f0 00 ]
X->H [ 00 07 00 0a 00 29 ff 00 10 ]

H->X [ 00 0d 00 0a 00 2a ff 00 01 0c 02 00 05 f0 00 ]
X->H [ 00 07 00 0a 00 2a ff 00 10 ]

H->X [ 00 0d 00 0a 00 2b ff 00 01 0c 02 00 06 f0 00 ]
X->H [ 00 07 00 0a 00 2b ff 00 10 ]

H->X [ 00 0d 00 0a 00 2c ff 00 01 0c 02 00 07 f0 00 ]
X->H [ 00 07 00 0a 00 2c ff 00 10 ]


Service State Configure
(Bring V5 IDs 0000, 0001, 0002, 0003 IN SERVICE):
H->X [ 00 0d 00 0a 00 2d ff 00 01 2c 02 00 00 f0 00 ]
X->H [ 00 07 00 0a 00 2d ff 00 10 ]

H->X [ 00 0d 00 0a 00 2e ff 00 01 2c 02 00 01 f0 00 ]
X->H [ 00 07 00 0a 00 2e ff 00 10 ]

H->X [ 00 0d 00 0a 00 2f ff 00 01 2c 02 00 02 f0 00 ]
X->H [ 00 07 00 0a 00 2f ff 00 10 ]

H->X [ 00 0d 00 0a 00 30 ff 00 01 2c 02 00 03 f0 00 ]
X->H [ 00 07 00 0a 00 30 ff 00 10 ]
```

V5.2 Example Call Processing Trace

LE side:

V5 ID = 0x0001

Subscriber ID: 0x0001

AN side:

V5 ID = 0x0002

Subscriber ID: 0x0001

```
Outgoing Call from LE side with LE side releasing:

Route Control:
H->X [00 45 00 e8 00 00 ff 00 01 29 02 ff fe 03 02 1e 19 00 03 00 13
00 02 00 15 00 5e 00 08 00 02 00 00 00 00 00 01 00 0f 00 01 0b 03 00
28 00 04 00 01 00 01 03 00 29 00 0f 01 12 0c 00 0a 35 30 38 38 36 32
33 35 30 30]

Route Control Acknowledge:
X->H [00 1d 00 e8 00 00 ff 00 10 02 03 00 28 00 04 00 01 00 01 02 1e
09 00 01 00 39 00 03 00 00 03 ]

Release Channel:
H->X [00 11 00 08 00 00 ff 00 02 0d 03 00 00 03 0d 03 00 00 03 ]
X->H [00 07 00 08 00 00 ff 00 10 ]

Channel Released With Data:
X->H [00 16 00 69 00 07 ff 00 01 0d 03 00 00 03 01 03 00 28 00 04 00 01 00 01 ]
H->X [00 05 00 69 00 07 ff ]
Incoming Call to LE with Subscriber release:

Request For Service With Address Data:
X->H [00 26 00 2d 00 04 ff 00 01 0d 03 00 00 04 00 08 02 03 00 28 00
04 00 01 00 01 03 00 29 00 09 01 12 06 00 04 33 32 31 38 ]
H->X [00 0c 00 2d 00 04 ff 00 01 0d 03 00 00 04 ]

Generate Call Processing Event of Answer:
H->X [ 00 0d 00 ba 00 00 ff 00 01 0d 03 00 00 04 01 ]
X->H [ 00 07 00 ba 00 00 ff 00 10 ]

Channel Released With data:
X->H [ 00 16 00 69 00 08 ff 00 01 0d 03 00 08 04 01 03 00 28 00 04 00 02 00 01 ]
H->X [ 00 05 00 69 00 08 ff ]
```

V5.2 Additional Call Flows

This section contains additional V5.2 message flows.

Important! This release of V5.2 supports LE only.

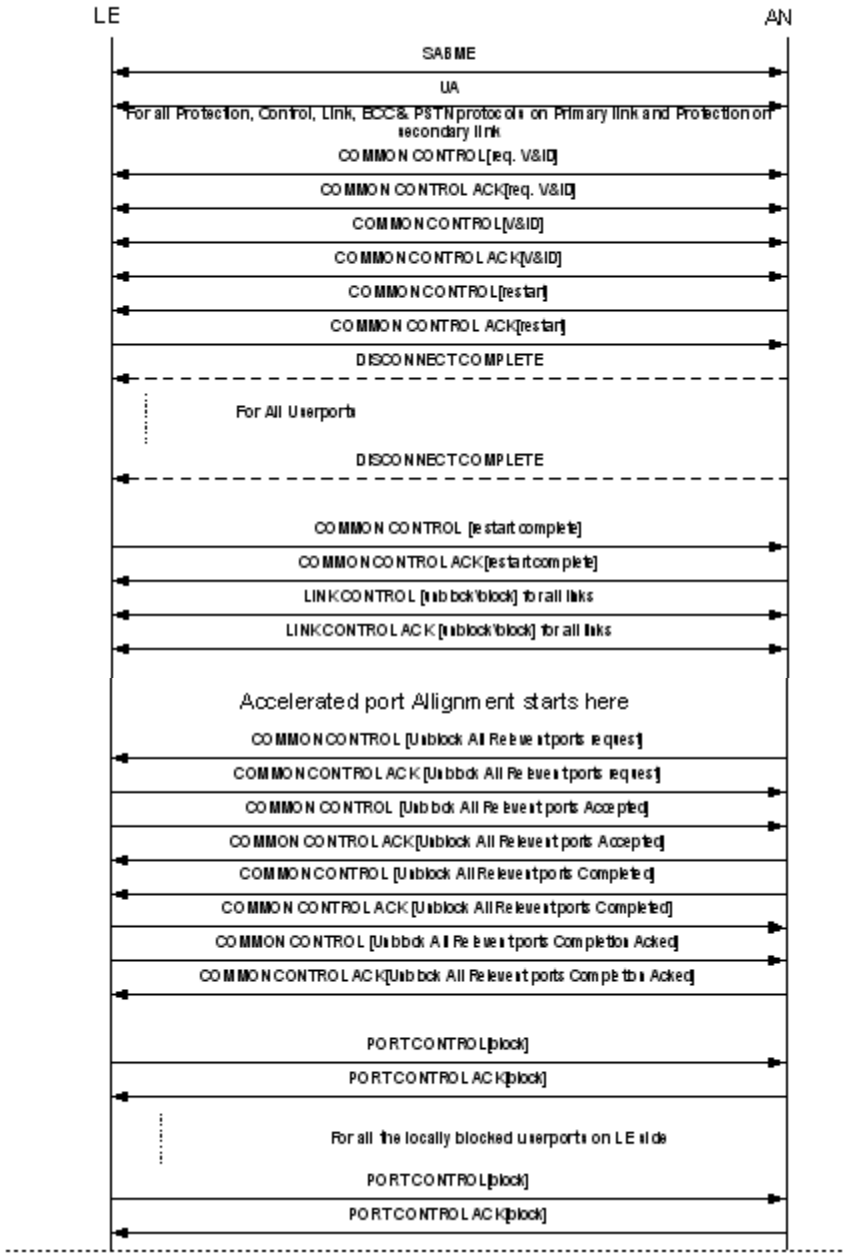
Important! AN is not supported for V5.2. Any text or illustrations (for example, call flows) indicating the AN side is for clarification purposes only.

Important! In the Startup procedures, DISCONNECT COMPLETE messages are shown as broken lines, indicating that these messages apply for a restart and for all cases when the interface goes down and comes back up.

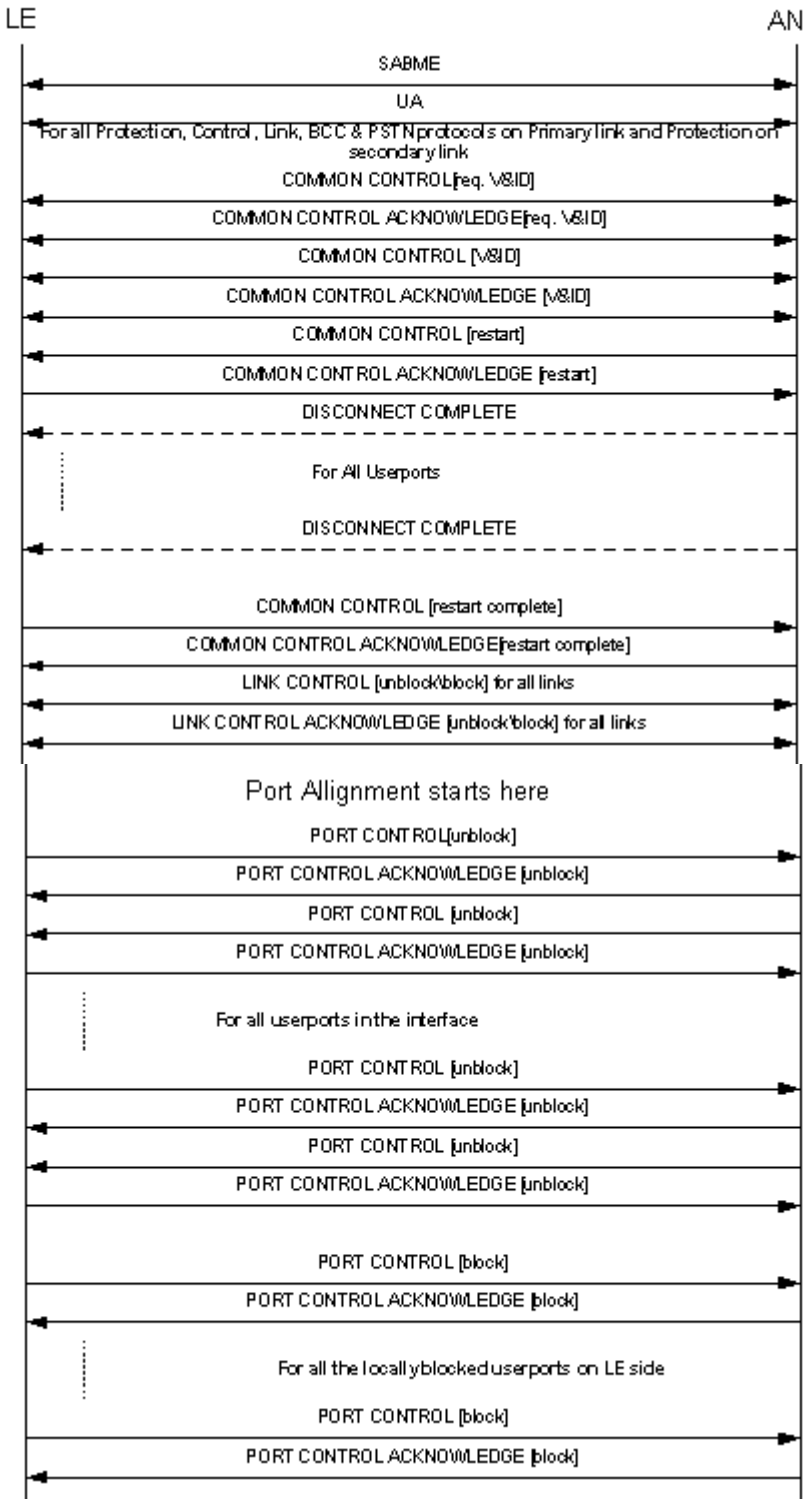
In all of the test cases, all PPL Config. Bytes were set to their default values, unless specified otherwise.

Test cases in which the third link is used can be applied to any link except Primary and Secondary links.

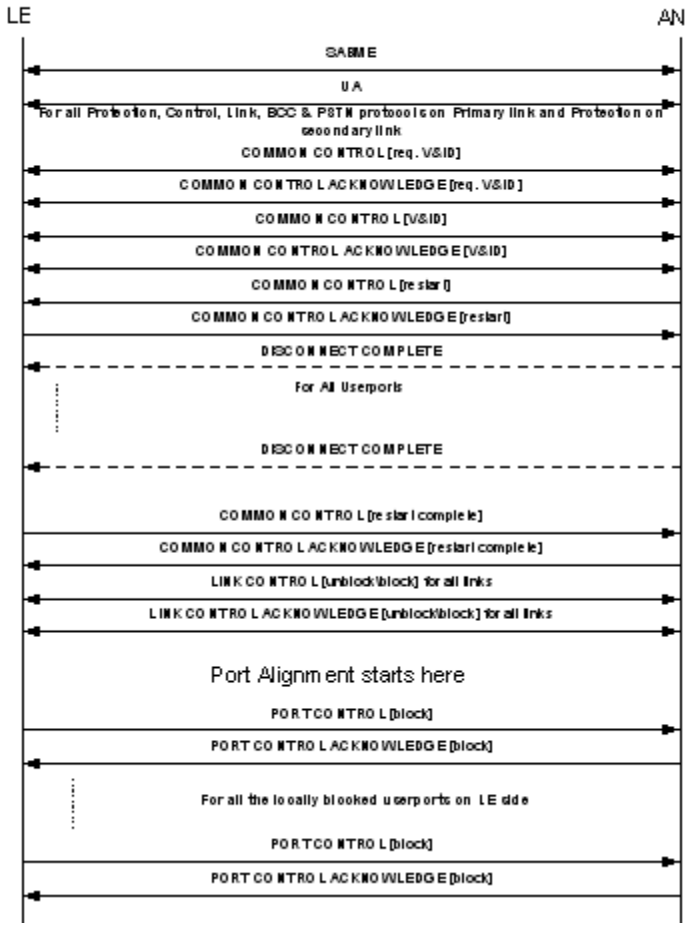
1. Startup Request with PPL Config. Byte #4 (Type of Port alignment) set to 1



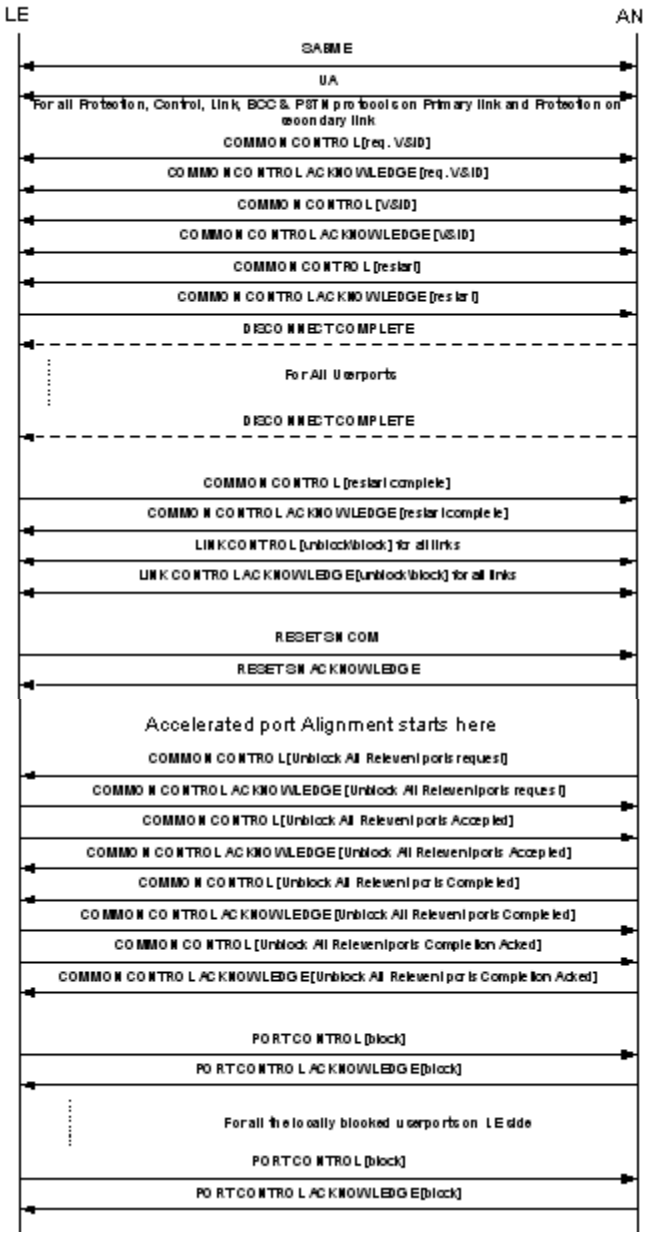
2. Startup Request with PPL Config. Byte #4 (Type of Port alignment) set to 0



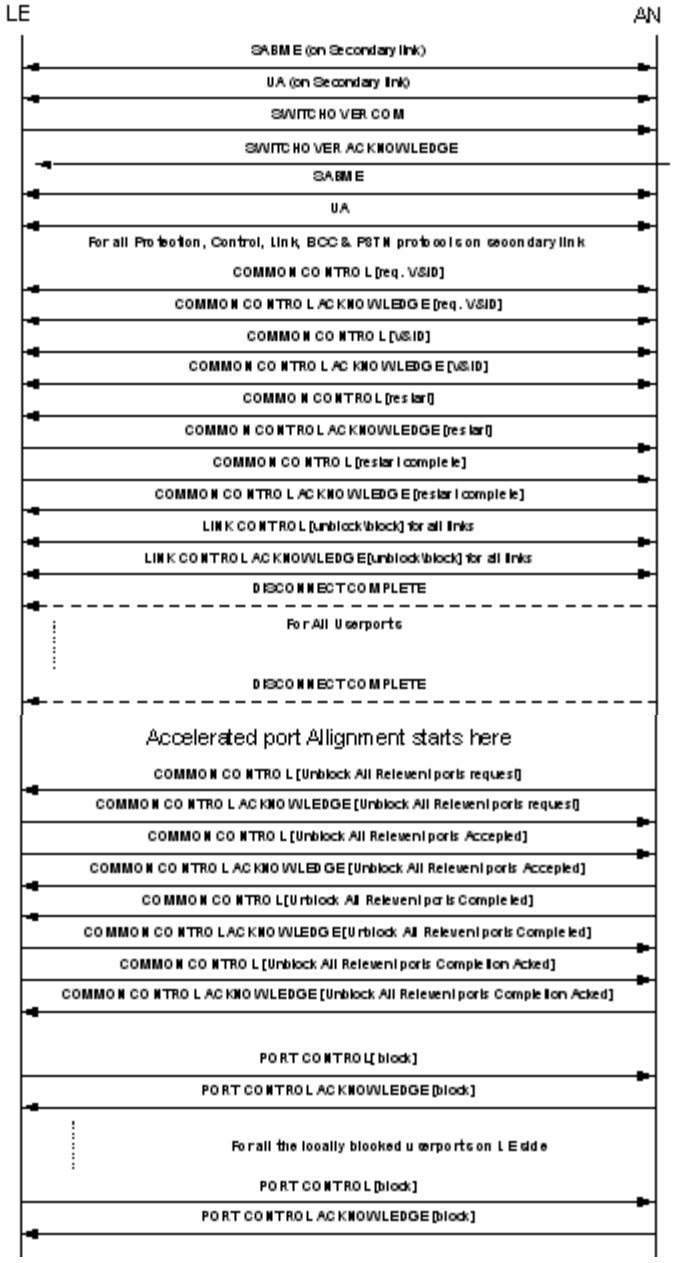
3. Startup Request with PPL Config. Byte #4 (Type of Port alignment) set to 0



4. Startup Request with PPL Config. Byte #5 (Reset Sequence Number) set to 1

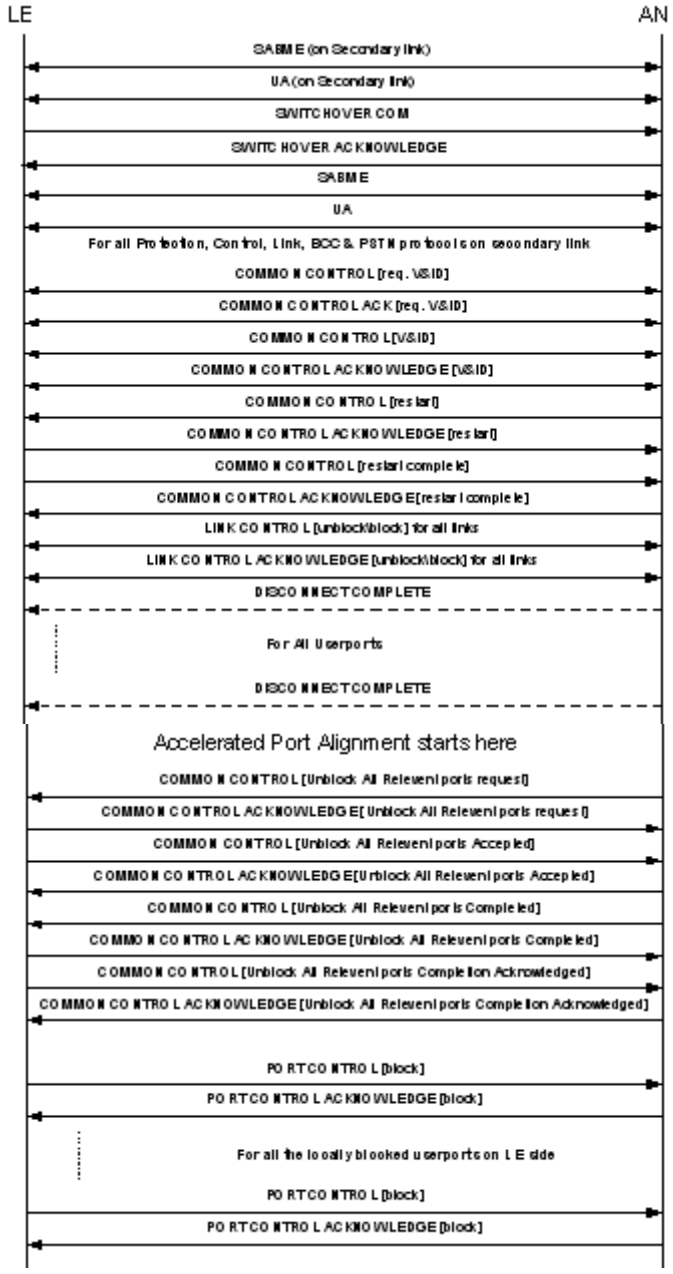


5. At Startup Primary and Secondary links are configured, but only secondary is available

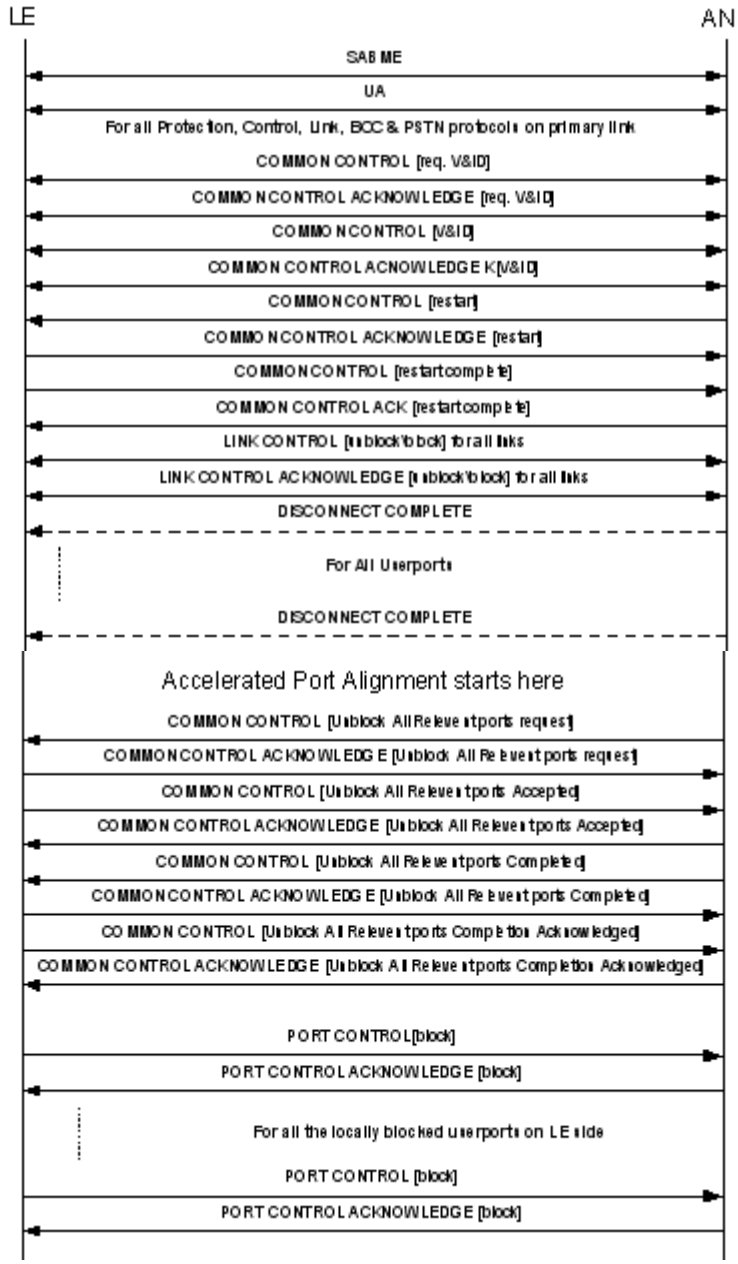


6. Primary and Secondary links are faulty, Restore Primary Link

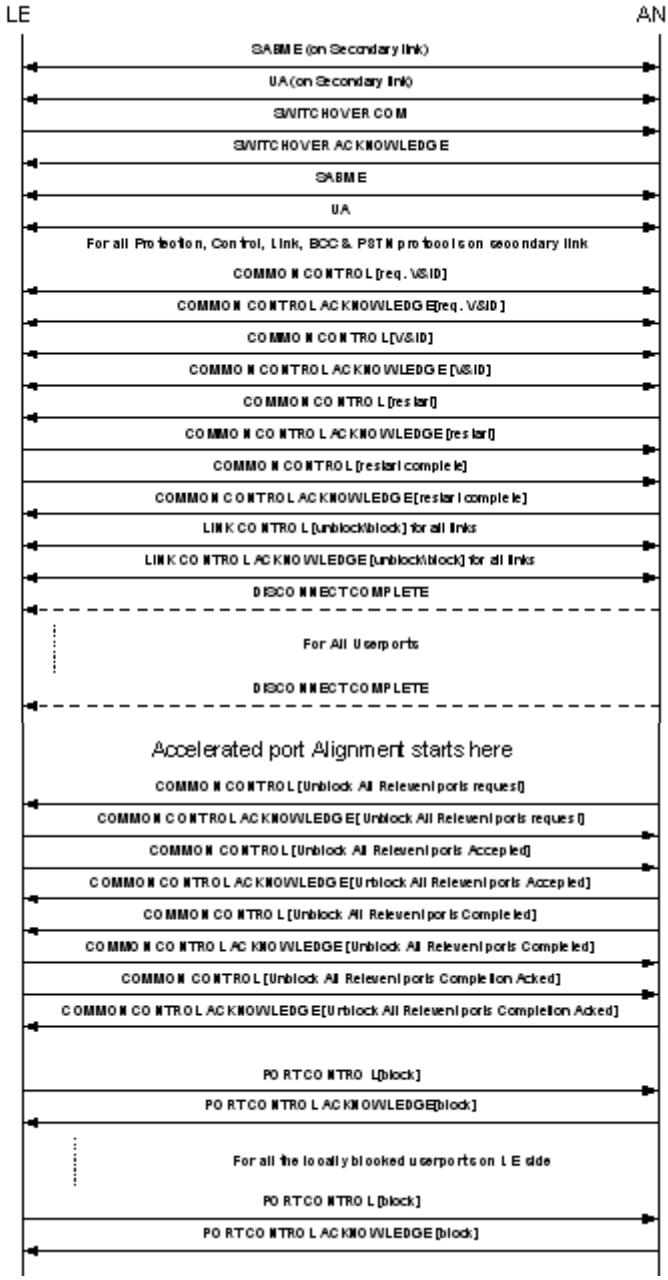
7. Primary and Secondary links are faulty, restore Secondary



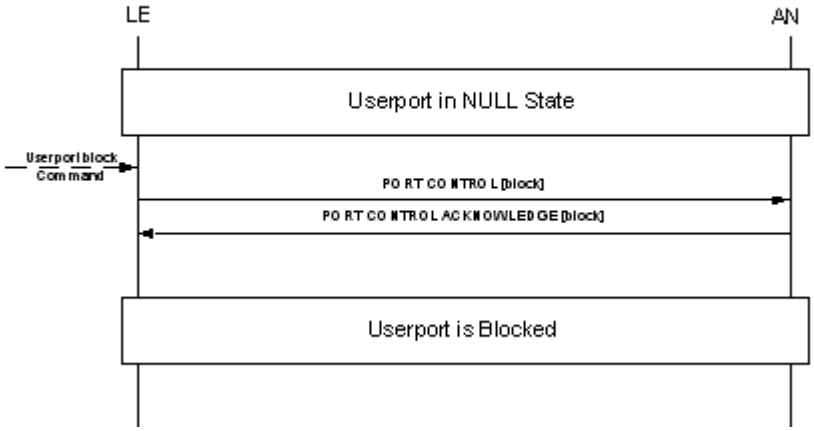
8. Secondary links is faulty and Primary Link Layer 2 is faulty, Layer 2 recovered from fault



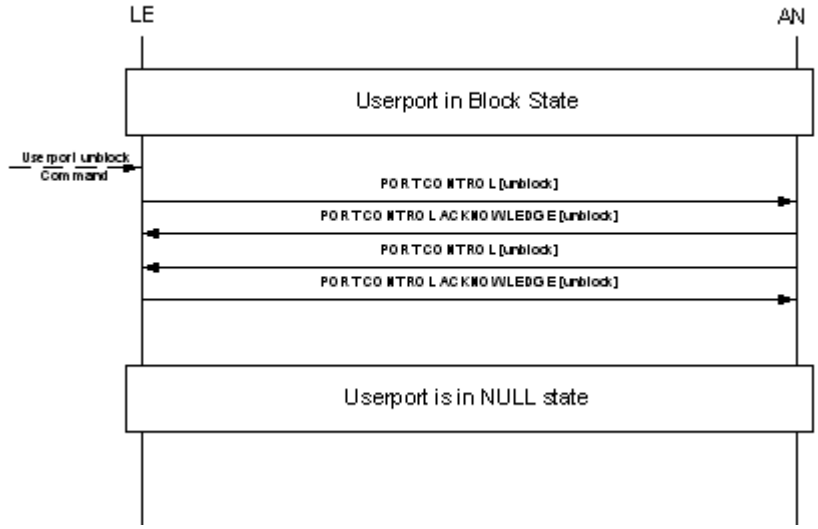
9. Primary is faulty and Secondary Link Layer2 is faulty, Layer2 recovered from fault



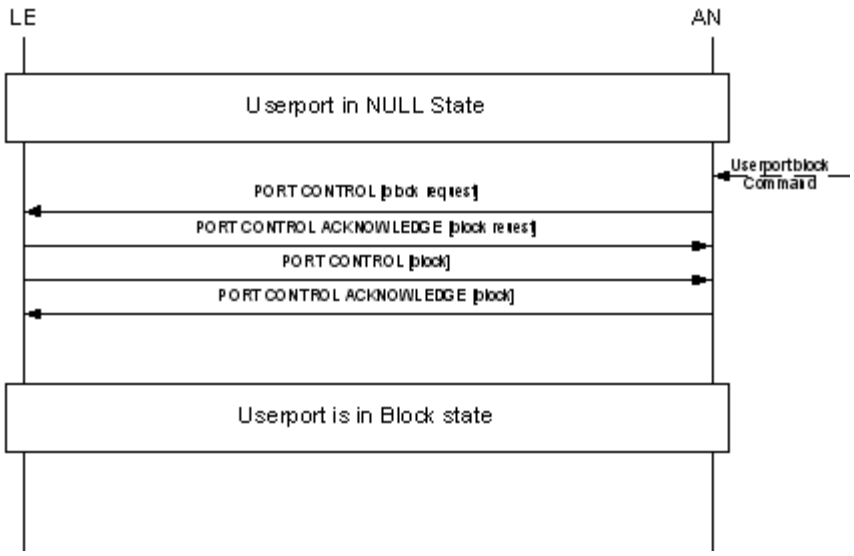
10. Userport blocking initiated by LE



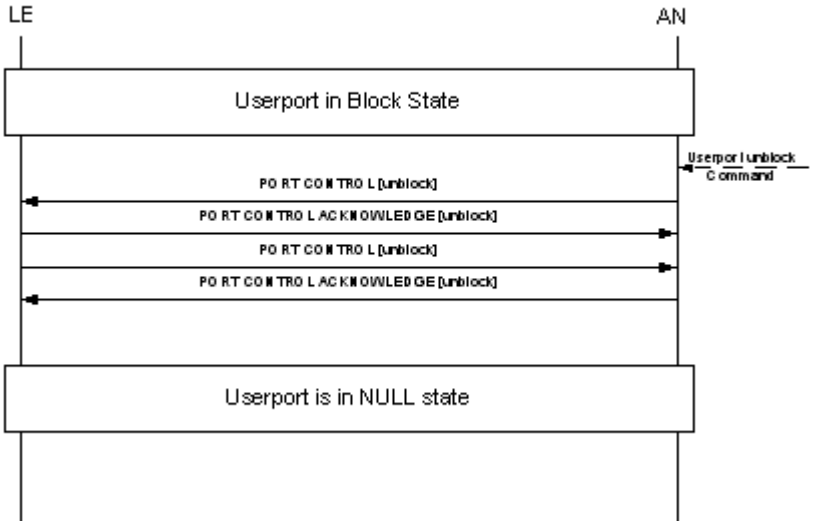
11. Userport unblocking initiated by LE



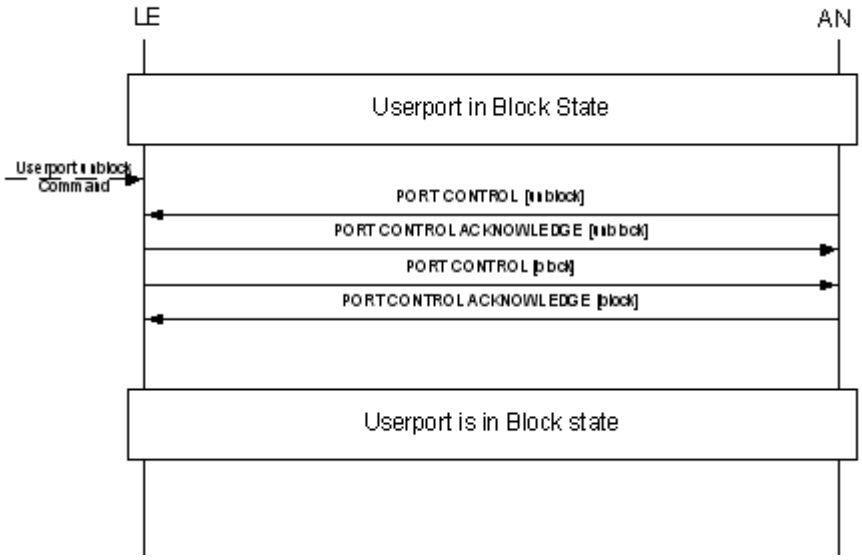
12. Userport blocking initiated by AN



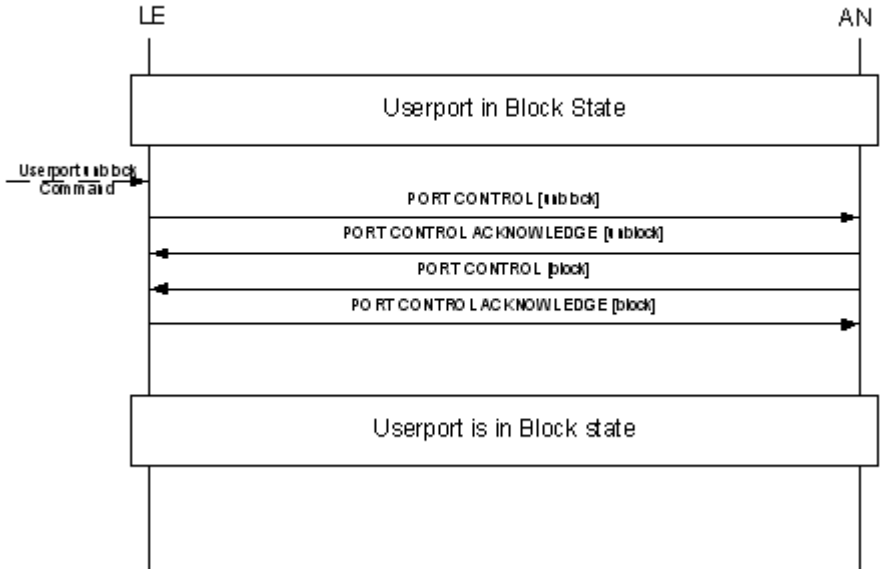
13. Userport unblocking initiated by AN



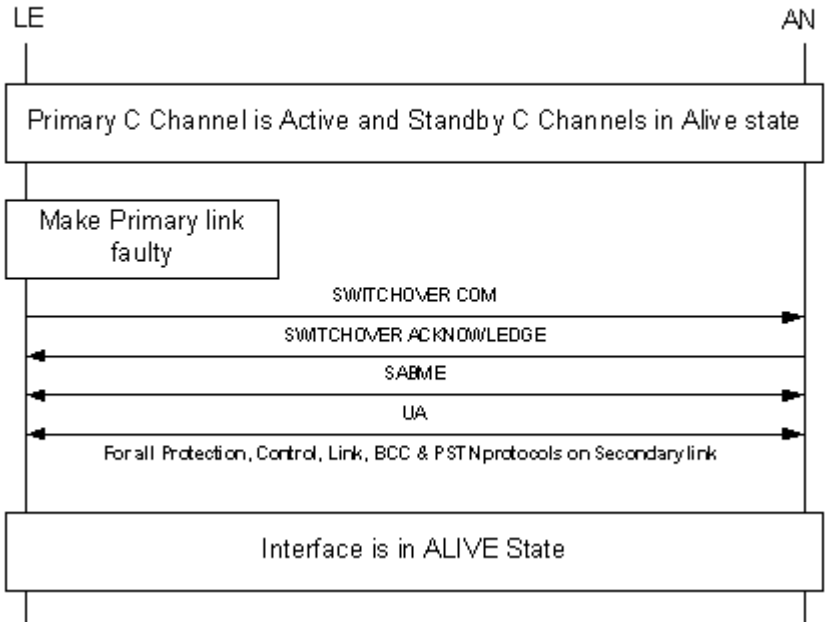
14. Userport unblocking initiated by AN and rejected by LE



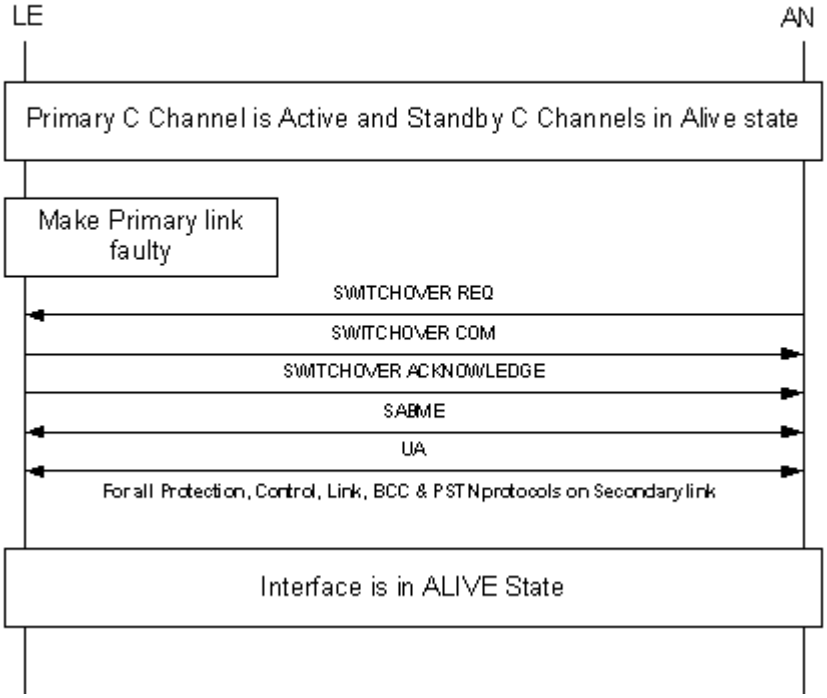
15. Userport unblocking initiated by LE and rejected by AN



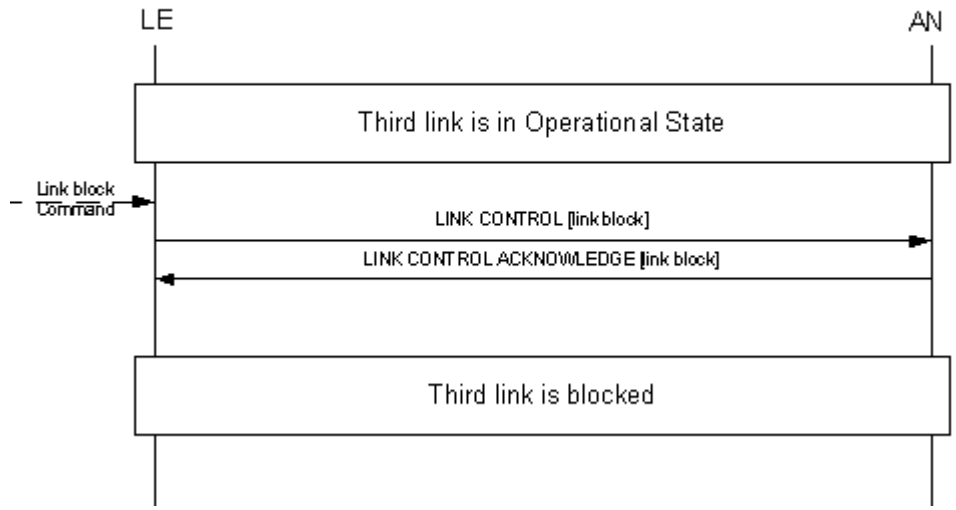
16. Protection Switchover initiated by LE



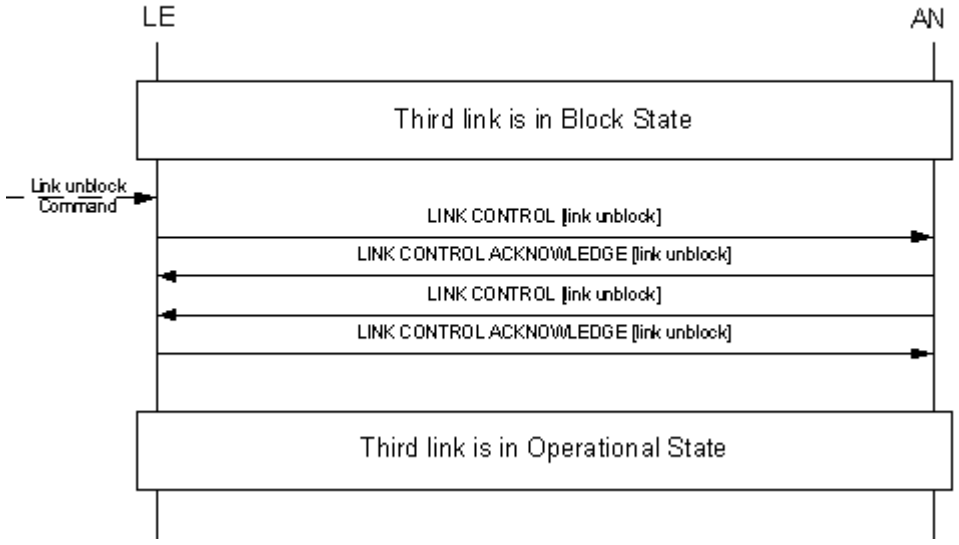
17. Protection Switchover initiated by AN



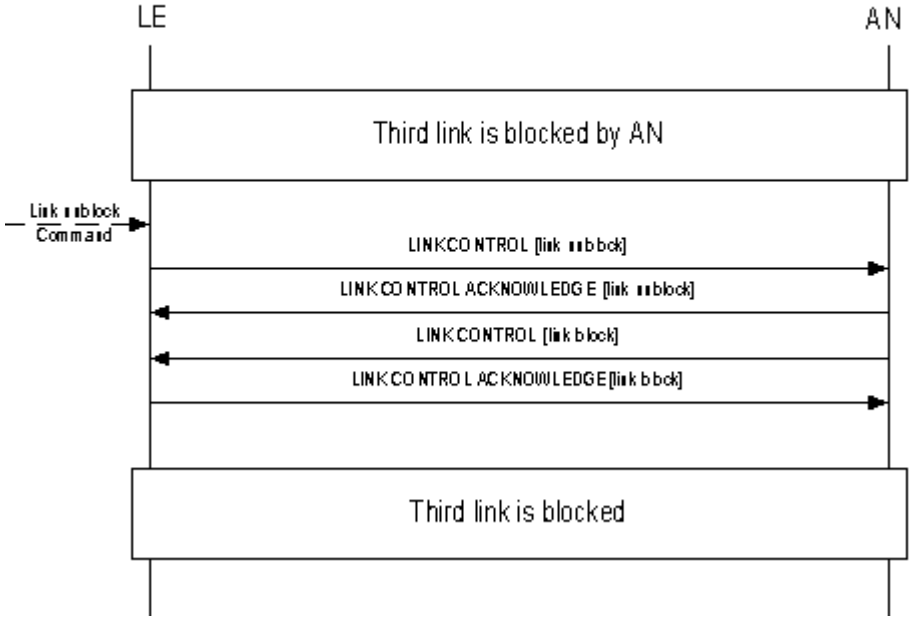
18. Link Blocking initiated by LE: The link includes NO Physical Channel



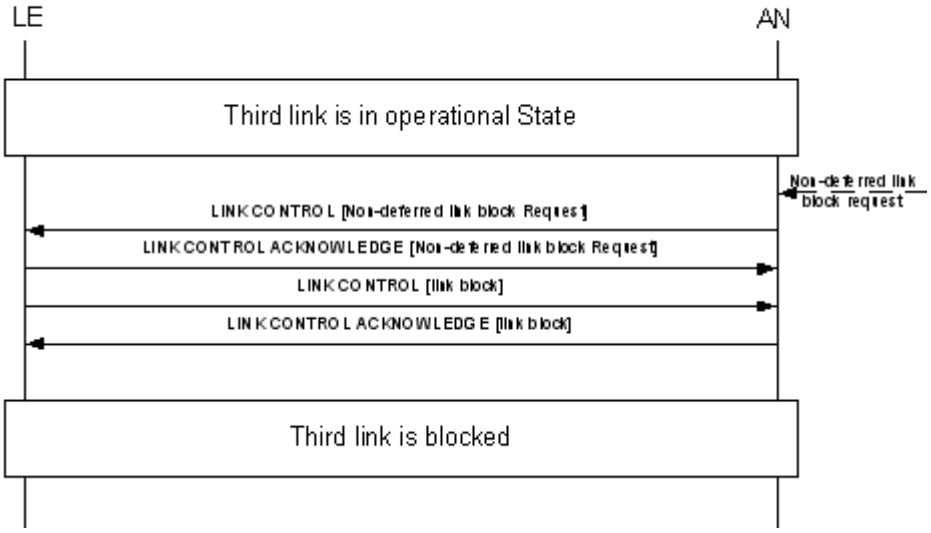
19. Link unblocking initiated by LE: The link includes NO Physical Channel



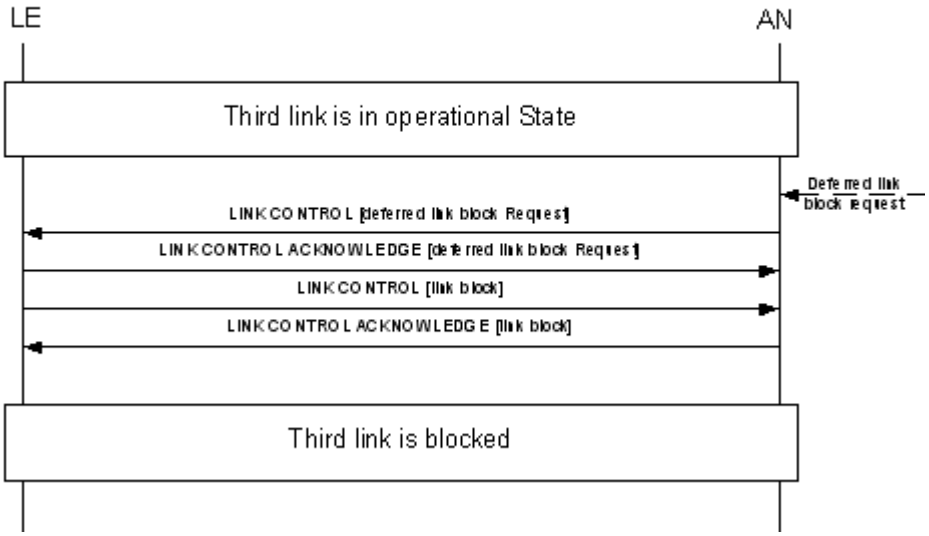
20. Link unblocking initiated by LE and Rejected by AN:
The link includes NO Physical Channel



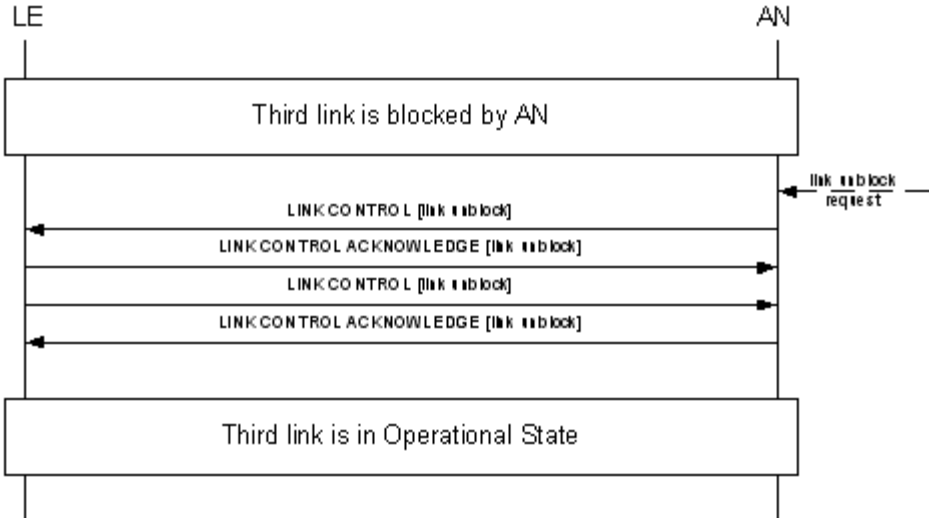
21. Non-deferred Link blocking initiated by AN:
The link includes NO Physical Channel and No Calls



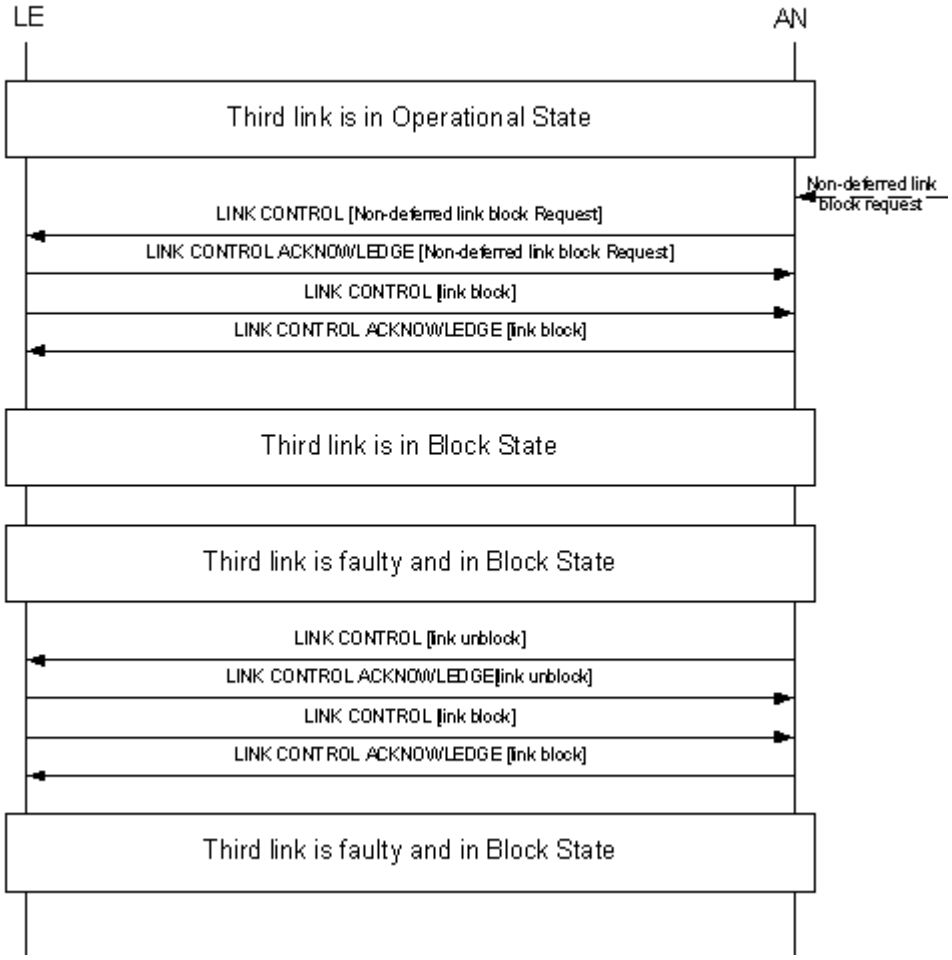
22. Deferred Link blocking initiated by AN:
The link includes NO Physical Channel and No Calls



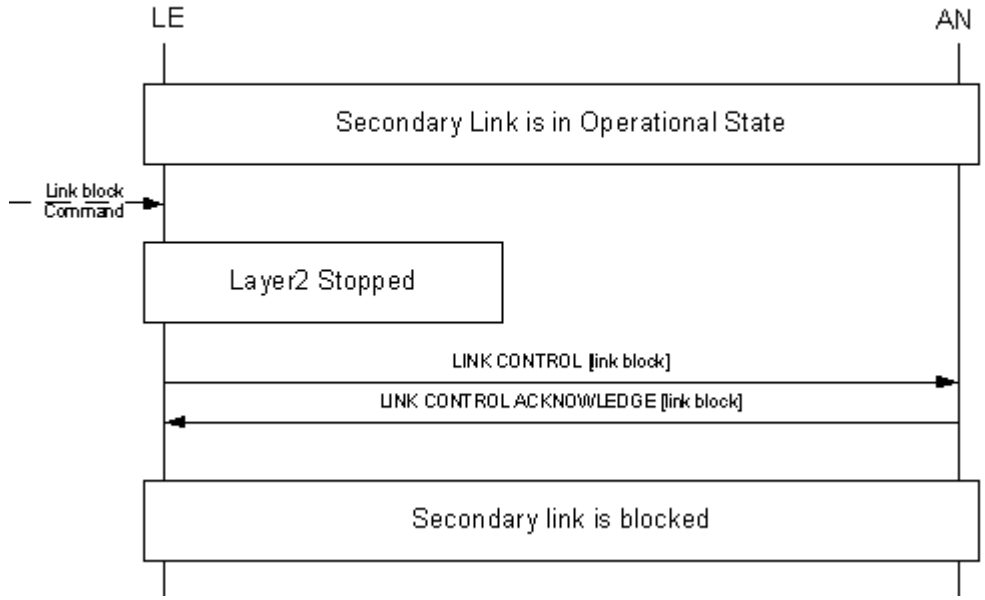
23. Link unblocking initiated by AN: The link includes NO Physical Channel



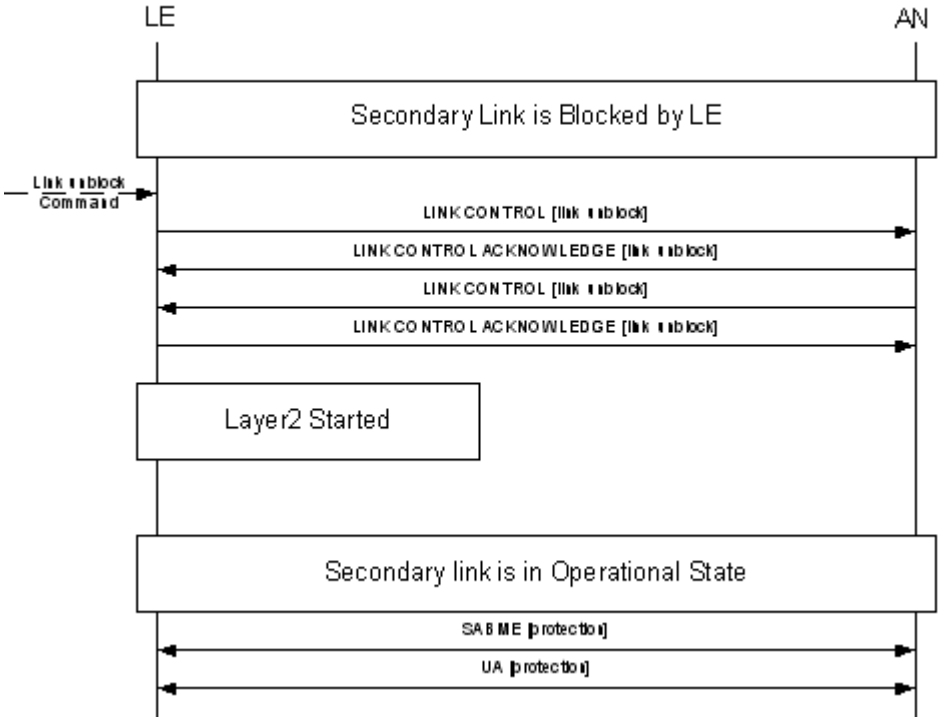
24. Link unblocking initiated by AN & rejected by LE because link is faulty:
The link includes NO Physical Channel



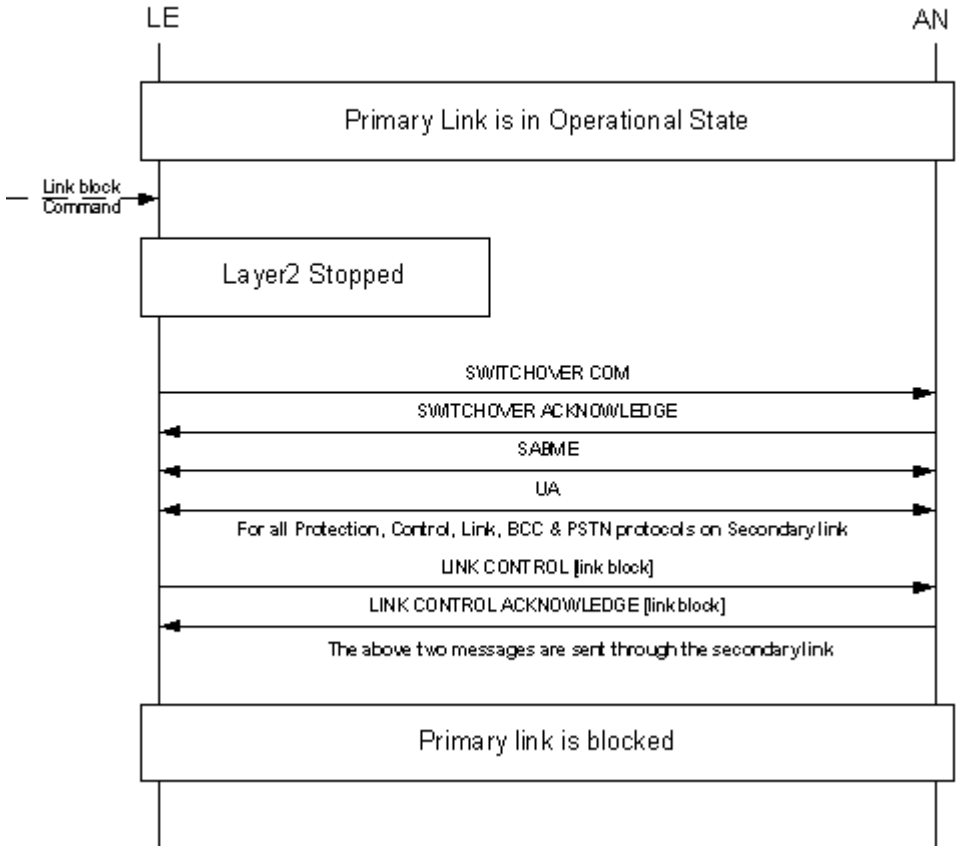
25. Link Blocking initiated by LE: The link includes Standby C Channel



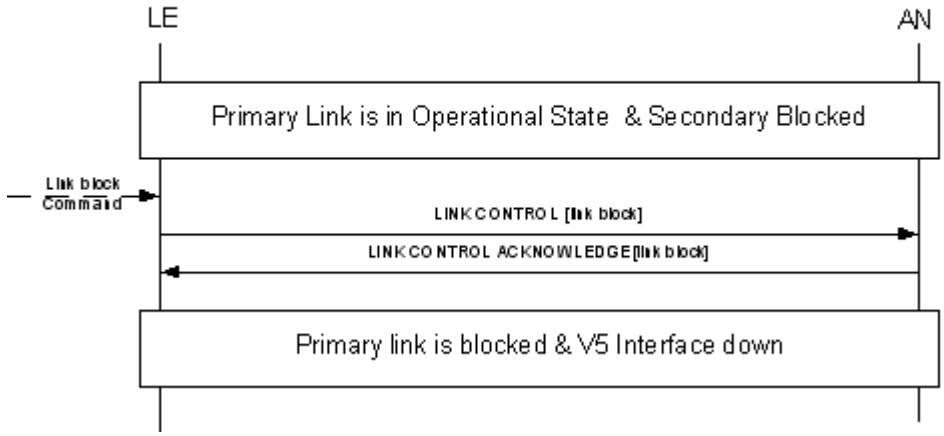
26. Link unblocking initiated by LE: The link includes Standby C Channel



27. Link Blocking initiated by LE: The link includes Active C Channel



28. Link Blocking initiated by LE & Secondary link is blocked & PPL Config. Byte#9 (Protective mode) set to '0': The link includes Active C Channel



29. Non-deferred Link blocking initiated by AN.
The link includes NO Physical Channel