



# **Dialogic® Converged Services Platform - Release 8.4.1 Engineering Release 3**

**Developer's Guide: Programmable Protocol Language**

# Copyright and Legal Disclaimer

---

Copyright © [1998-2008] Dialogic Corporation. All Rights Reserved. You may not reproduce this document in whole or in part without permission in writing from Dialogic Corporation at the address provided below.

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, Realblobs, Realcomm 100, NetAccess, Instant ISDN, TRXStream, Exnet, Exnet Connect, EXS, ExchangePlus VSE, Switchkit, N20, Powering The Service-Ready

Network, Vantage, Connecting People to Information, Connecting to Growth, Making Innovation Thrive, and Shiva, among others as well as related logos, are either registered trademarks or trademarks of Dialogic.

The names of actual companies and products mentioned herein are the trademarks of their respective owners.

# 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](http://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](mailto:techsupport@cantata.com) or call (781) 433-6900.

## **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 NEGLIGENCE), 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 publication 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.
<i>n</i>	The letter <i>n</i> is a generic placeholder for a number.
Sans serif mono space	Indicates a command name, option, input, output, non-GUI error, and system messages.
<i>Sans serif monospace italic</i>	Indicates a parameter name in an input message. Example: move *.dot a: c: -s The -s is the parameter.
<i>Serif italic</i>	Indicates the name of a book, chapter, path, file, or API message. Example: <i>UserDirectory/Config.exe</i>
<b>Boldface</b>	Indicates keyboard keys, key combinations, and command buttons Example: <b>Ctrl+Alt+Del</b>
<b>Sans serif boldface</b>	Identifies text that is part of a graphical user interface (GUI). Example: Go to the <b>Configuration</b> menu and select <b>Card-&gt;Span Configuration</b>

# Contents

Copyright and Legal Disclaimer .....	i-2
Dialogic Product Line Warranty .....	i-4

---

## **1 PPL Introduction**

Basics of PPL .....	1-2
PPL Environment .....	1-6
PPL Guidelines .....	1-8
API Messages used for Protocol Modification .....	1-12
State Machines .....	1-14
PPL Timers .....	1-20
PPL Auditing .....	1-22

---

## **2 Common Atomic Functions**

---

## **3 T1 Atomic Functions**

---

## **4 E1 Atomic Functions**

---

## **5 Layer 4 Atomic Functions**

Call Control Channel Management (0x0061) .....	5-2
Layer 4 Call Management (0x0062) .....	5-68
Layer 4 Physical Connection (0x0063) .....	5-83
Call Control Router (0x0064) .....	5-85

---

## **6 SS7 Atomic Functions**

L3P CIC (0x000F) .....	6-2
L3P Link (0x0010) .....	6-21

ISUP CPC (0x0012) .....	6-24
ISUP BLS, HGBS MGBS .....	6-46
ISUP SPRC (0x0013) .....	6-47
ISUP SSC (0x0085) .....	6-66
L3P TUP (0x0011) .....	6-71
L3P BT IUP (0x0011) .....	6-73
BT IUP CPC (0x0052) .....	6-112
SCCP SCRC (0x0066) .....	6-135

---

## **7 ISDN Atomic Functions**

L3P Call Control (0x0005) Atomic Functions.....	7-2
L3P B Channel Control (0x0007) Atomic Functions.....	7-23
L3 Call Reference (0x0008) Atomic Function .....	7-25
L3 Global Call Reference (0x0009) Atomic Function .....	7-29
L3 D Channel Control (0x000A) Atomic Functions .....	7-31
L3P PSTN (0x0091) Atomic Functions .....	7-39
L3P BCC (0x0092) Atomic Functions.....	7-58
L3P MGR (0x0093) Atomic Functions .....	7-68

---

## **8 VDAC/IP Network Interface Atomic Functions**

# 1 PPL Introduction

**Purpose** This chapter describes the PPL environment, terms and acronyms related to PPL, symbols used in the PPL Composer, timers, auditing and provides examples of how PPL components can be changed.

## Basics of PPL

---

### Benefits of PPL and the PPL Composer

Dialogic's Programmable Protocol Language (PPL) provides the ability to easily access and modify software operations on a per PPL component basis, giving the user full control over call processing. The PPL software load resides on system cards and consists of a state machine engine, default protocol state machines (\*.ppl files) and atomic function/event libraries (\*.exl files).

A PPL protocol is an association of a state/event table and a primitive table. The state machine engine uses the tables to drive PPL objects through various states in response to internal or external events.

You can customize protocols using the PPL Composer application, which provides a graphical representation of PPL state machines. Protocol variants can be developed easily by modifying default protocols that can be obtained from the Dialogic protocol library.

PPL protocols are independent of switch system software, allowing for the creation of new protocols without upgrading system software. This allows for quick changes required to manage network signaling variants. Protocols are created with high-level Atomic Functions (AF), abstracting the user from detailed implementation logic.

Multiple protocols can be defined for each PPL component and selectively assigned to each object being managed.

### Terms and Acronyms

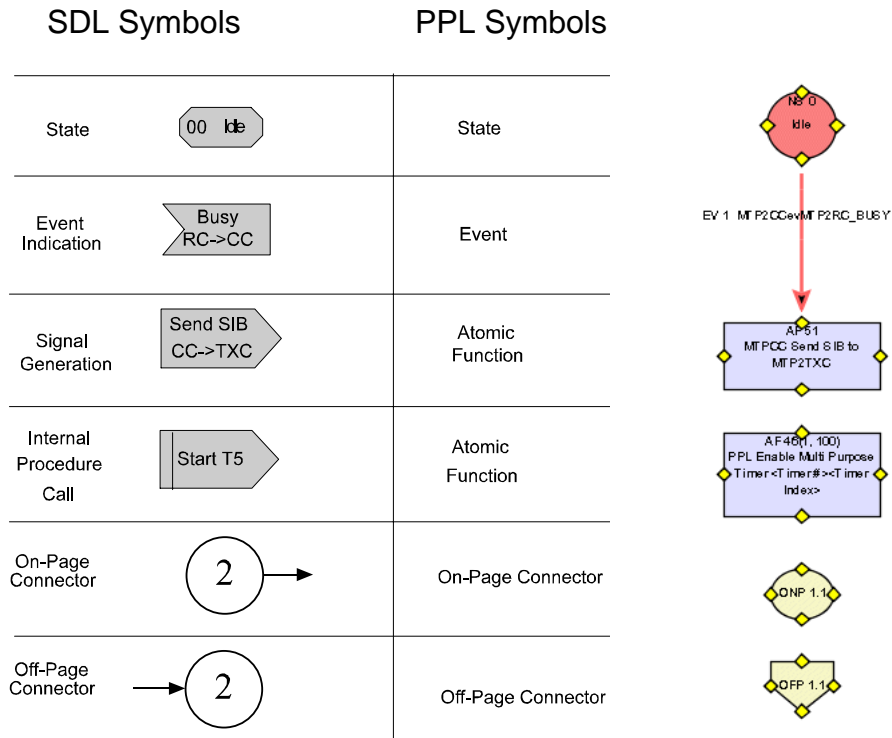
You should familiarize yourself with the following terms and acronyms used throughout this manual. Each is discussed in more detail within the manual.

AF	Atomic Function. A state machine software routine which normally performs one independent task.
*.ppl	File extension for PPL protocol data file used by the PPL Composer application.
Event	A condition that, when met, drives a state machine from one state to another.
*.exl	File extension for files containing the atomic function/event library for a PPL component.
INS	In Service
L3	Layer 3 (Network Signaling Protocol Layer)

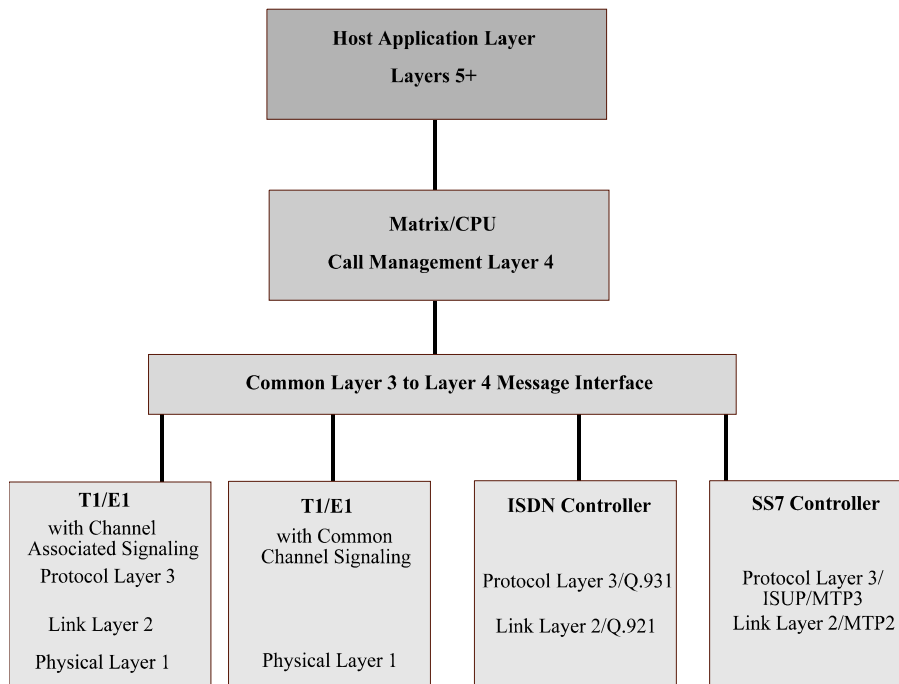
L3P	Layer 3 Plus (interface between L3 and L4)
L4	Layer 4 (Call Processing Layer)
L5	Layer 5 (Host Application Layer)
OOS	Out of Service
PPL	Programmable Protocol Language. An Environment which allows for different software components within Dialogic's switching platforms to be externally programmed and downloaded by the host application.
PPL Component	A programmable software component that uses PPL state machines that can be modified with the PPL Composer.
PPL Object	A system object that is controlled by a PPL state machine such as a channel, an ISDN D channel, or an SS7 link.
Primitive	A list of atomic functions invoked by a particular event.
Primitive Table	A list of all primitives used in a particular state machine.
Protocol	A state machine comprised of a state/event table and a primitive table.
State	The current "context" of a PPL object, such as Idle, Seizing, Answered.
State/Event Table	A table of all states used in a state machine and their associated events/primitives.

**Symbols** PPL state machines use symbols to represent states, atomic functions, and events. These correspond to standard Specification and Description Language (SDL) symbols as shown in the figure below. The PPL symbol for a state that is internal is denoted with an "I" in front of the "S".

**Figure 1-1 State Machine Symbols**



**Software Architecture Figure 1-2 CSP Software Architecture**



## PPL Environment

---

### Overview

The PPL environment is designed into the switch system software, integrating custom PPL-generated state tables with system software residing on PPL-controlled components on each card. The developer has the ability to determine which call processing events are processed by the switch and how the switch responds to the events.

### API messages manage PPL components

PPL components are configured and managed using the PPL API messages. The API messages allow you to modify PPL timers, Config Bytes, transmit signaling (E1), event indications and requests, and more.

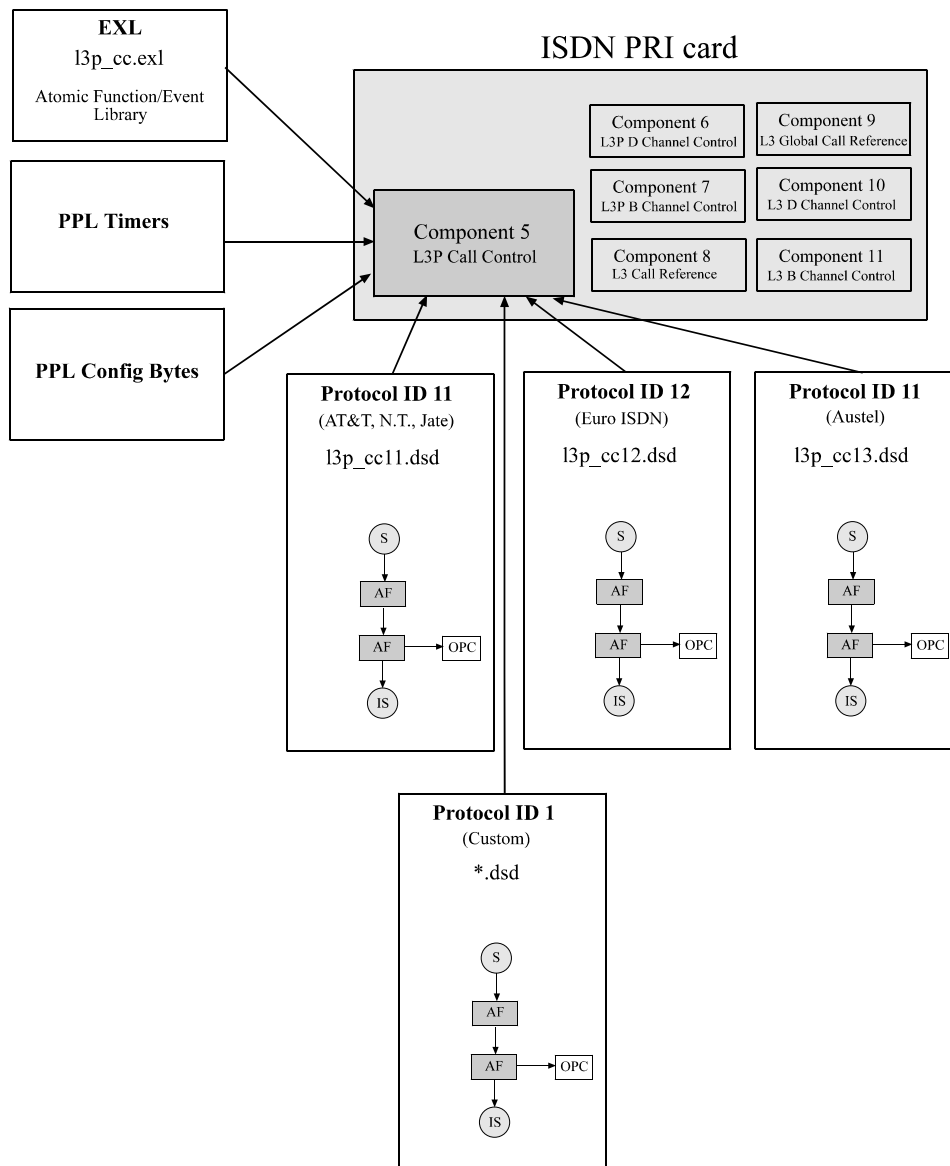
### PPL Composer

All components are configurable with the PPL Composer, allowing you to modify protocol state machines to conform to variants. The default \*.ppl and \*.exl files for components that are released for custom programming with the CSP system software. If you require modification of unreleased PPLs, contact Dialogic Technical Support.

In addition to the default protocols, up to 10 custom protocols can be stored per component and assigned on a per object basis. The default protocols can be modified using the PPL Composer.

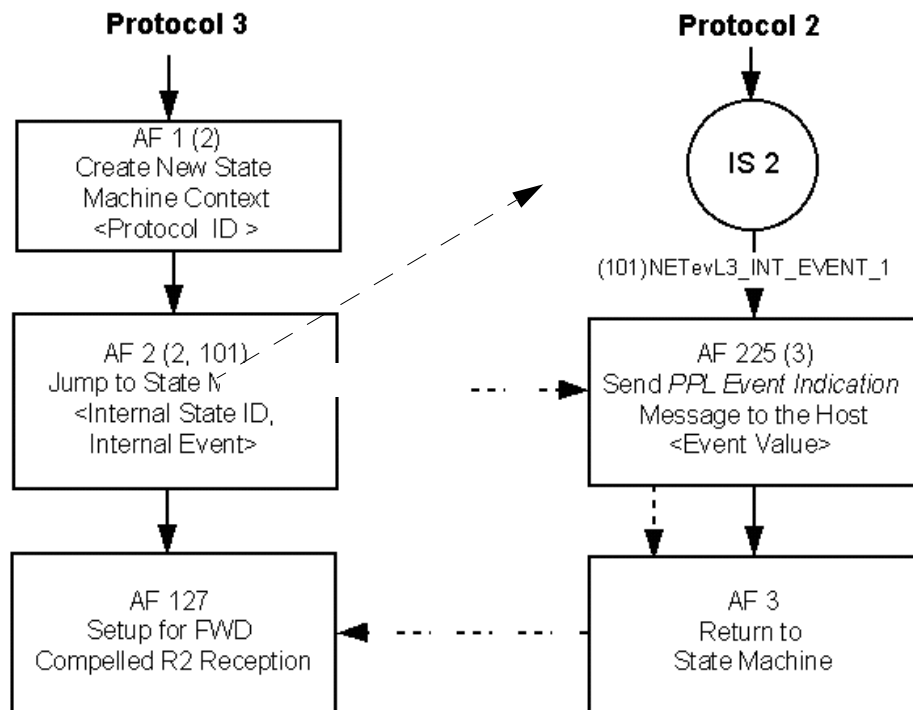
Each PPL component has a default protocol (or protocols), timers, Config Bytes, and EXL (atomic function/event library). The timers and Config Bytes are programmable using API messages.

**Example** *Figure 1-3, PPL Environment (1-7) uses the ISDN PRI card to illustrate the PPL environment. The Layer 3 Plus Call Control component (5) has three default protocol variants for the three connection types supported by the ISDN PRI card (Protocol IDs 11, 12, and 13). All three variants use the same timers, Config Bytes, and EXL. Using the PPL Composer, another protocol could be created (Protocol ID 1-10) and downloaded to the switch, and selectively assigned to objects.*

**Diagram Figure 1-3 PPL Environment****Switching between protocols**

The ability to switch between several PPL protocols is provided by AF 2, which invokes a primitive resulting in the channel using a new set of protocol tables. It is the responsibility of the “secondary” protocol to return to the primary protocol (protocol assigned to the channel).

**Figure 1-4 Switching Between Protocols**



## PPL Guidelines

---

**Overview** This section gives general guidelines to follow when developing protocols. For more detailed information on using PPL for specific protocols, consult the *API Reference* and developer's guides.

**Idle State** The Idle state must be consistent with the default protocol for a component. A protocol must transition to the primary protocol without having any knowledge of the primary protocol's state/event and primitive table data.

### T1 and E1

Since the E1 OOS protocol has no knowledge of the primary T1 and E1 protocol, the transmit IDLE line signaling ABCD bit values must be configured by sending a *PPL Transmit Signal Configure* message. This

should be done prior to bringing the channel In Service. The Out of Service line signaling value is transmitted when a channel is being controlled by the OOS protocol (channel ID out of service).

**Internal States** A given Internal State (test state) must accept all possible Internal PPL Events (test results) from a test AF that is invoked. The description of all test Atomic Functions indicate the possible Internal PPL Events that may be returned.

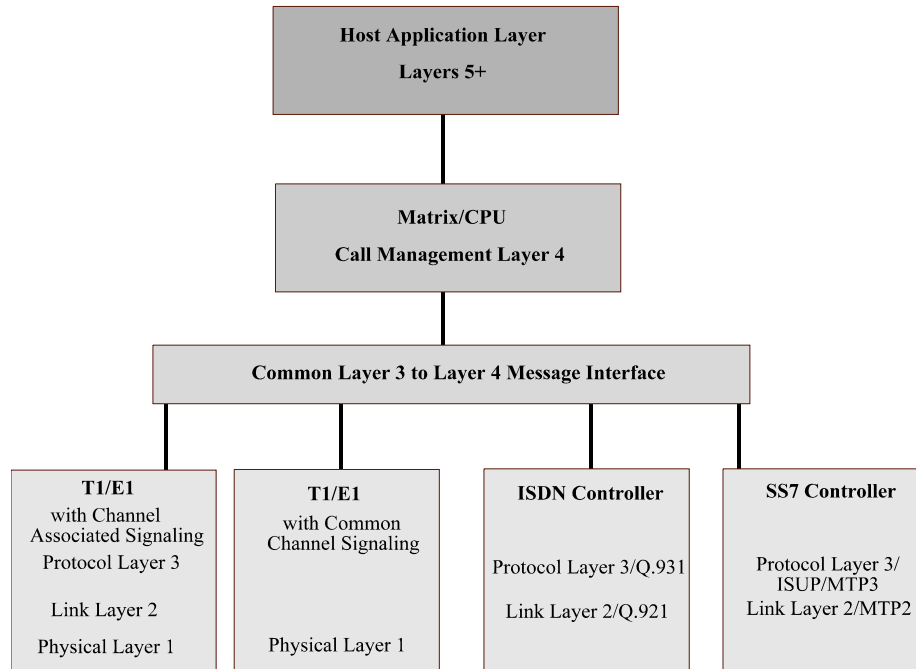
**Multi-Purpose PPL Timers** Each component has 3 multi-purpose timers available for use within a protocol. These timers can be activated through an “Enable Timer” AF passing the timer number (1-3) in Arg1 and the preconfigured PPL timer setting ID (1-100) in Arg2. See the Overview section for more details on PPL Timers.

The preconfigured timer settings can be changed by the host using the *PPL Timer Configure* message. As timer adjustments are independent from the protocol tables you can modify a timer value without having to create new protocol tables.

When a timer expires, an event is generated into the PPL state machine for the associated channel. The timer events are numbers 191, 192, and 193.

**Multiple Resident Protocols** The CSP allows for up to 10 custom protocols per component to be resident in the system at one time. Custom protocols can be given Protocol IDs from 1 to 10. Dialogic default protocols begin at Protocol ID 11.

**Call Processing** The CSP software architecture shown in the figure below is similar to the OSI Layered Model. Inter-Layer communications between Layer 3 and Layer 4 must take place for call management. Specific events and AFs are provided for each PPL component to manage call processing. This section describes the required Layer 3 to Layer 4 messaging for incoming and outgoing calls.

**Figure 1-5 CSP 2000 Software Architecture****Incoming Call**

Condition	L3 to L4 Message
Incoming Call Setup Completed	L3 ⇒ L4 Setup Indication by L3
Terminating Party being Alerted	L4 ⇒ L3 Alerting
Terminating Party Answered	L4 ⇒ L3 Connect

**Outgoing Call**

Condition	L3 to L4 Message
Initiation of Outgoing Call	L4 ⇒ L3 Call Request or L4 ⇒ L3 <i>Outseize Control</i>
Outseizure Acknowledgment Successfully Detected by L3	L3 ⇒ L4 Alerting
Answer Detected by L3	L3 ⇒ L4 Connect

**Network-initiated Call Release**

Condition	L3 to L4 Message
Network Release Detected	L3 $\Rightarrow$ L4 Disconnect by L3
L4 Acknowledgment of L3 Disconnect Request	L4 $\Rightarrow$ L3 Clear
L3 Network Release Completed (L3 in Idle state)	L3 $\Rightarrow$ L4 Clear

**Host/Associated Party-initiated Call Release**

Condition	L3 to L4 Message
L4-initiated Release	L4 $\Rightarrow$ L3 Clear
L3 Network Release Completed (L3 in Idle state)	L3 $\Rightarrow$ L4 Clear

# API Messages used for Protocol Modification

---

**Overview** PPL Protocols can be modified in two ways:

- PPL Composer

Default PPLs can be modified using the PPL Composer to create protocol variants. See the *PPL Composer User's Guide*.

- API Messages

PPL API messages can be used to modify a number of PPL configurations without modifying the protocol state machines.

## **API Messages** PPL Table Download Initiate and PPL Download

*PPL Table Download Initiate* and *PPL Download* messages are used to download protocol tables to the switch.

### **Downloading and Assigning a Protocol**

Download the protocol to the switch. See the developer's guides for downloading methods. Assign the protocol to channels or other objects (such as SS7 links) using the *PPL Assign* message.

Perform any required PPL configuration such as:

- Timers (*PPL Timer Configure*)
- Config Bytes (*PPL Configure*)
- Signaling (*PPL Transmit Signal Configure*, E1 only).

Bring the channels or other objects in service using the Service State Configure message. The following API messages are used for PPL configuration. See the *API Reference* for complete information on each message.

### **PPL Create and PPL Assign**

These messages are used to create a protocol using the downloaded tables and to assign the protocol to PPL-controlled objects.

### **PPL Configure**

This message is used to modify a component's PPL Config Bytes. The default Config Bytes are documented in the *API Reference*.

**PPL Data Query**

This message is used to retrieve information on Config Bytes, PPL Timers, and the state of a component.

**PPL Event Indication and PPL Event Request**

These messages are used to generate PPL events between the host and the switch.

**PPL Audit Configure and PPL Audit Query**

These messages are used to configure and retrieve audit data on a card. See the *PPL Auditing (1-22)* in this chapter for more information.

**PPL Timer Configure**

This message is used to modify a protocols timer values. See *PPL Timers (1-20)* for more information.

**PPL Transmit Signal Configure**

This message is used to configure transmit line signaling on E1 channels.

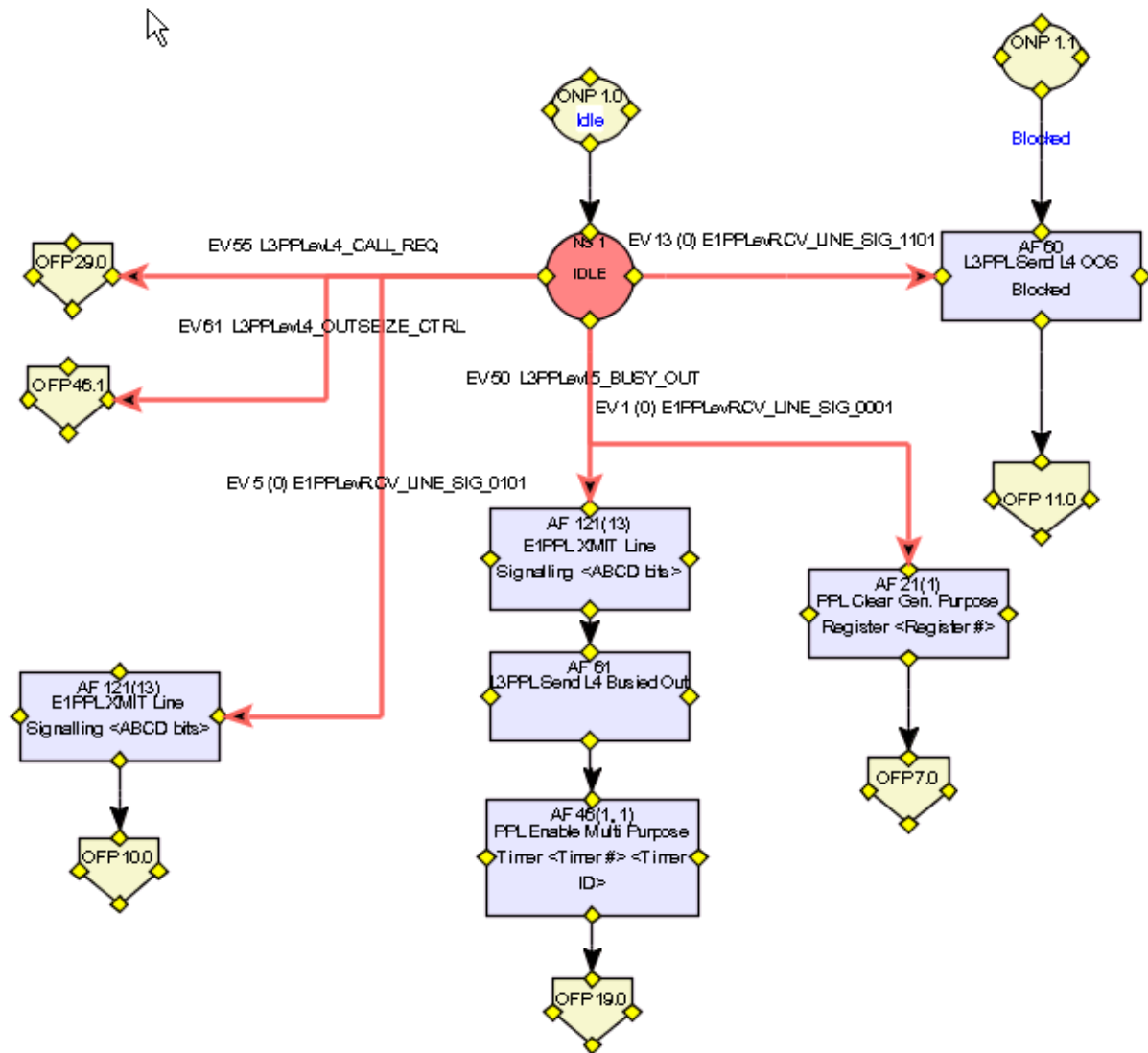
**PPL Protocol Query**

This message is used to generate a report on the custom protocols downloaded to a PPL component.

# State Machines

**Overview** The figure below shows an example of a state machine. The state machine engine drives each channel through its protocol tables based on the occurrence of an event. When a normal state is reached, processing of the event is complete. The dynamic generation of internal blocking states is also supported for service resource requests such as digit receiver allocation.

**Figure 1-6 State Machine Example**



- State** The current “context” of a component. There are three types of states:
- Normal
  - Blocking
  - Internal

### **Normal State**

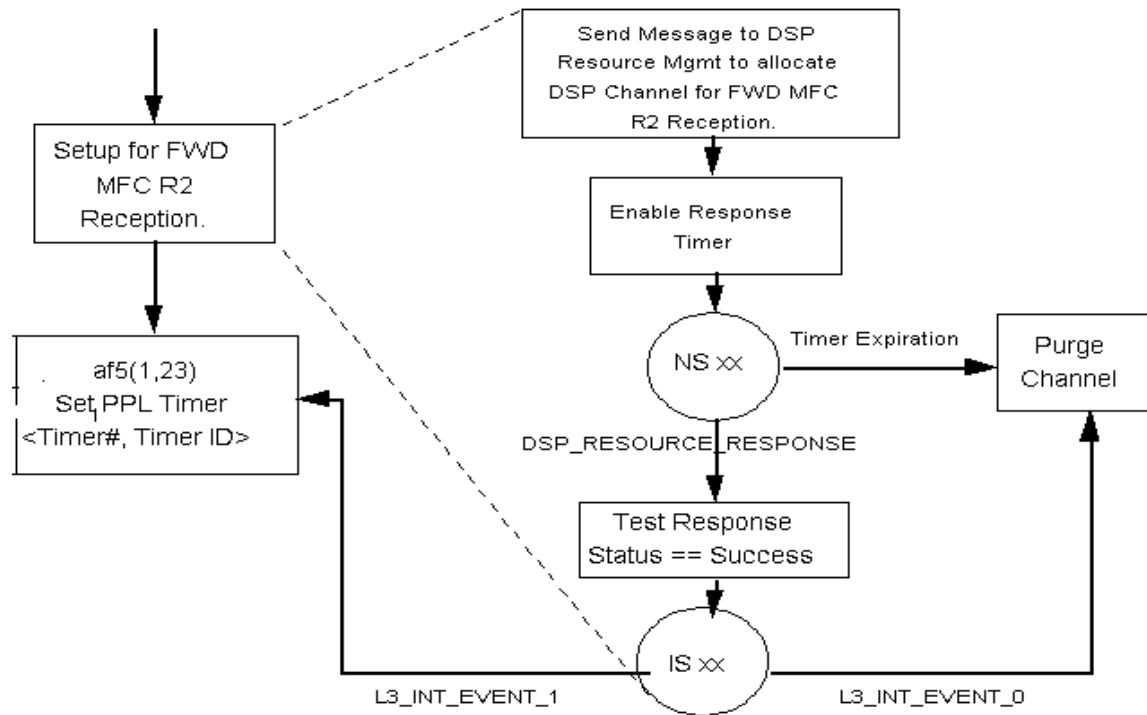
A normal state can be a wait state (i.e., waiting for a line signaling event, or host input), or a stable state (i.e., conversation state).

### **Blocking State**

A blocking wait state is automatically generated by the PPL when a “blocking” AF is invoked. A blocking AF is used for allocation of any off-board service resource required by the PPL. Upon the receipt of a positive result of the resource request, the channel “unblocks” and resumes the remainder of the primitive. Blocking is performed on a per object basis without affecting other objects.

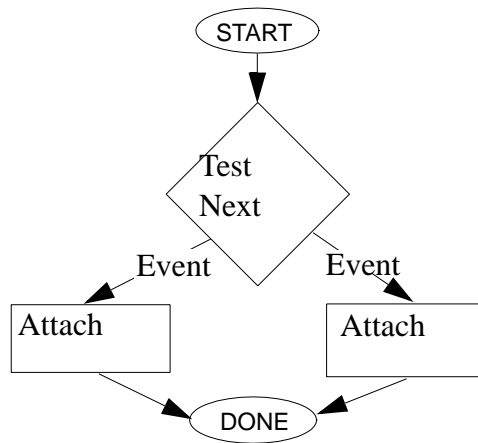
Blocking States are transparent to the PPL designer, but they can be tracked in a PPL audit. A blocking state is indicated by 0xFF in the State Status field of PPL Audit Data (see *PPL Auditing (1-22)*). This isolates the protocol developer from the details of the internal service management provided by the CSP when creating new PPL protocols. Only normal and internal states can be programmed by the host.

An example of a Blocking state is shown in the figure on the next page.

**Figure 1-7 Blocking State****Internal State**

Internal States are used to take a decision branch based upon performing a logical test. An Internal State is initiated by a Test AF, which will result in an internal event being returned based upon the result of the test. Internal states only accept internal events. The documentation for test AFs shows the internal events generated for each result.

The format of an internal state can be represented in standard logic flow chart form, as shown in *Figure 1-8, Logic Flow Chart Decision Branch (1-17)*. The PPL representation of the flow chart as an internal state is shown in the figure on the next page.

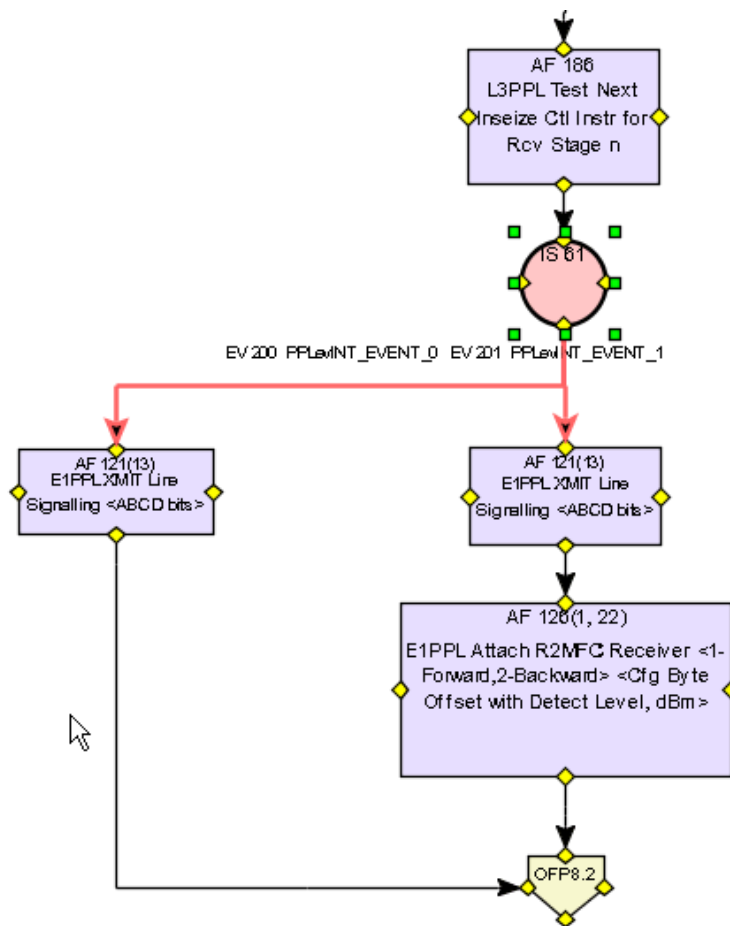
**Logical Flow Chart  
Decision Branch****Figure 1-8    Logic Flow Chart Decision Branch**

The next page shows a PPL state machine representation of the logic flow chart decision branch above.

### PPL State Machine Decision Branch

Figure 1-9

### PPL State Machine Representation of Logic Flow Chart Decision Branch



**Event** A condition that drives a component out of its current state by invoking an associated primitive, such as:

- Receive Line Signaling Change
- Dialtone Detected
- Alerting Message

**Atomic Function (AF)** A predefined, two-argument function that performs a simple protocol action, such as:

- Transmit Line Signaling
- Send Layer 4 Disconnect
- Allocate Digit Buffer

The AFs for all default protocols that can be modified using the PPL Composer are documented in the section for each card. They include a description, argument values and ranges, internal events returned for tests, and the minimum software version.

**Primitive** A grouping of one or more AFs and their arguments that allows multiple actions and tests to be initiated in response to a PPL event.

# PPL Timers

---

**Overview** There are two types of timers available to a protocol state machine:

- Generic Protocol Timers
- Protocol-specific Timers

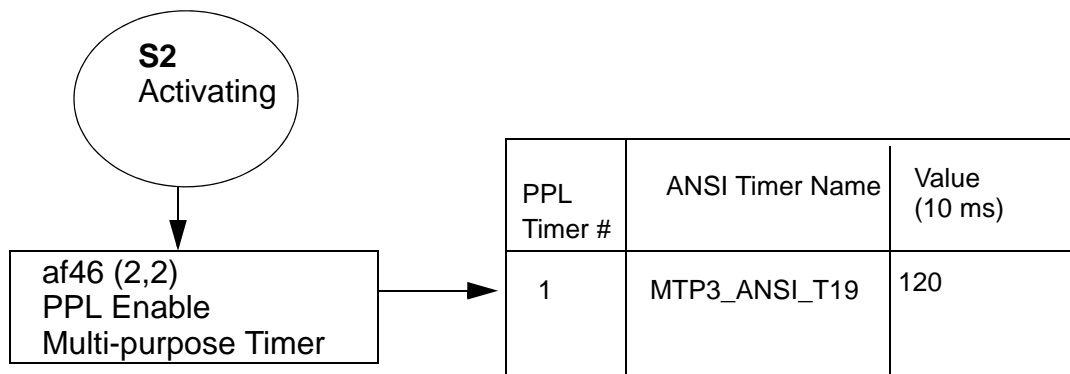
**Generic PPL Timers** Each component has access to a list of 100 timer values that are enabled in a protocol using AFs 46, 47, and 48. Up to three timers can be enabled at any one time. Each component using timers has a table that contains the value of specific timers. Argument 2 of the Generic PPL Timer AFs is an index into the component's timer table where the value for the required timer is located.

## Example

The figure below shows the use of a timer AF in the MTP3\_LSAC state machine. AF 46 initiates one of the PPL timers (in this case, Timer 2, as defined by Argument 1 of the AF). Argument 2 is the index into the component's timer table. In this case, Argument 2 indicates Index 2 in the timer table, which is the MTP3\_ANSI\_T17 timer with a default value of 1s.

See the *API Reference* for the default values of each component's Generic PPL Timers.

**Figure 1-10 Generic Protocol Timers**



**Protocol-Specific Timers**

The E1 component uses various timers when transmitting and receiving signaling tones or dial pulses. They are as follows:

- R2 Signaling
- DTMF Signaling
- MFR1 Signaling
- Dial Pulse

The default timer values are documented in the *PPL Timer Configure* message, which is used to modify any of the timer values. See the “*API Reference*” for more information.

**Modifying Protocol Timers**

You can modify all of the timers and values in the PPL Timer tables using the *PPL Timer Configure* message. This allows the timer values to be modified without modifying the component state machine. To modify a components Generic Protocol Timer list, indicate a Timer Type of Generic Protocol Timer (0x01), the Timer ID (index into the components timer table), and the new Timer Value. To modify an E1-specific Signaling Timer, indicate the Timer Type, Timer ID (see message), and new Timer Value.

**Example API format**

The following example shows how to modify the MTP3\_ANSI\_T17 timer from one second (default) to two seconds.

Byte	Field Description	Value and Indication
0	Frame Character	0xFF
1	Length (MSB, LSB)	0x0011
2		
3	Message Type (MSB, LSB)	0x00CF (PPL Timer Configure)
4		
5	Reserved	0x00
6	Sequence Number	0xSN
7	Logical Node ID	0xFF
8	AIB	0x00 (Single Entity)
	Address Method	
9	Number of Address Elements	0x01

Byte	Field Description	Value and Indication
10	Address Type	<b>0x09</b> (SS7 Link)
11	Data Length	<b>0x02</b>
12	Data[0] Stack ID	<b>0x01</b>
13	Data[1] Link ID	<b>0x01</b>
14	PPL Component ID (MSB, LSB)	<b>0x002E</b> (MTP3_LSAC)
15		
16	PPL Timer Type	<b>0x01</b> (Generic Protocol Timer)
17	PPL Timer ID	<b>0x02</b> (MTP3_ANSI_T17)
18	PPL Timer Value (MSB, LSB)	<b>0x00C8</b> (2 s)
19		
20	Checksum	<b>0xCS</b>

## PPL Auditing

---

**Overview** When PPL Auditing is enabled on a card, all state machine activity on the card is recorded in an audit log on either a per-PPL component basis or an individual entity basis. A log consists of a number of audit entries, each representing a state transition on a specific entity (channel). The length of the audit entries varies per PPL component.

Under busy call conditions, the user has the option initiating PPL Auditing on an Entity Basis instead of PPL Auditing on a Component Basis. This option provides the user with more accurate error auditing. Enabling the auditing function on an entity basis allows the auditing buffer to store only the information pertaining to user-selected entities. When auditing on a component basis, the ability to determine the cause of a problem can be lost. This is because the multi-entities that overload the auditing buffer can be overwritten due to a large volume of incoming voice traffic.

**IMPORTANT:** Before initiating the PPL Audit function, contact Dialogic Technical Support for information on selecting the Audit Type options in the PPL Audit Configure and PPL Audit Query messages.

### **PPL Audit Configure**

PPL Auditing is enabled on a per-card basis using the *PPL Audit Configure* message. This message accepts and validates all AIBs that correspond to the provided component ID. A message received with a Slot AIB will be processed as in previous releases. A message received with an AIB, other than a Slot AIB, will be processed as an Individual Entity Audit configuration message. The Audit Type field is a bit mask that provides bits 0 through 3 to enable or disable PPL Auditing features on a component or entity basis. For example, setting bit 0 enables PPL auditing. See the *PPL Audit Configure* message for the Audit Type information.

There are two new bit options in the Audit Type field. Bit 3 allows configuration of individual entities, and bit 2 enables the configuration. Both bits can be set at the same time or bit 2 can be enabled or disabled at any time to activate or deactivate individual auditing.

To support PPL Auditing on an Entity basis, the PPL Audit Configure message supports AIBs that are associated with the following PPL components:

- E1 (0x0001)
- T1 (0x0003)
- ISDN L3P Call Control (0x0005)
- ISDN L3 Call Reference (0x0008)
- SS7 L3P CIC (0x000F)
- SS7 ISUP CPC (0x0012)
- Channel Management (0x0061)
- Call Management (0x0062)

### **PPL Audit Query**

The *PPL Audit Query* message supports Slot AIB, and all AIBs that are associated with the PPL components. This message supports the querying of all PPL Audit configurations. For example, when this occurs a single byte of Audit configuration data will be returned. This byte will contain the current settings for board level PPL Auditing, board level PPL Error Alarm settings, and Individual PPL Auditing.

If the AIB specifies an entity other than a Slot AIB, the byte will also contain entity level PPL Auditing and entity level Error Alarm settings. For example, this information will be queried by setting the Audit Type to 0x02 in the *PPL Audit Query* message. The *PPL Audit Query* message Audit Types are as follows:

- An Audit Type of (0x00) requests the PPL Audit information for the specified entity.
- An Audit Type of (0x01) requests a PPL ERROR audit on a per card basis.
- An Audit Type of (0x02) requests the current PPL Audit configuration. It will return a PPL Audit Query response containing the status, the AIB, the component ID, the Audit Type, and an Audit Configure byte containing the current Audit Configuration setting. The slot AIB will be supported when the Audit Type is Audit Configuration (0x02). The bits (0 through 4) of the Audit Configuration byte are listed in the *PPL Audit Query* message.

**Audit Data** The exact data in an audit log varies per audit type and PPL component. See the *PPL Audit Query* message for the exact data returned. The following information is included in all audit entries:

- Protocol ID
- State Status
- Protocol-specific Information
- PPL Event
- Initial State
- Next State
- Error Status
- Timestamp

**Retrieving an Audit Log** An audit log can be retrieved by sending a *PPL Audit Query* message. There are two audit query types: PPL State Transitions and PPL Errors. PPL State Transitions are queried on a per-PPL component/entity basis. All state transitions for a specified PPL component on a specified entity are reported. PPL Errors are queried on a per PPL component/card basis. All PPL errors for the specified PPL component on the specified card are reported.

An entire log cannot always be retrieved with one message. Audit entries are grouped into audit blocks. When you query a log, you specify audit block 0 in the first message, audit block 1 in the second

message, and so on until all audit entries have been returned. When there are no audit entries remaining, the response status to the *PPL Audit Query* message will indicate 0x00D7 (End of PPL Audit Data).

**PPL Error Alarm** By enabling the PPL Error Alarm, the host will be notified with an *Alarm* message when a PPL Error occurs on an entity. The PPL Error Alarm is enabled by setting bit 1 of the Audit Type field in the *PPL Audit Configure* message. The *Alarm* message will indicate Entity - 0x04 (Channel) and Alarm # - 0x03 (PPL Error). The Alarm Information fields contain the audit entry data from the PPL Audit log.



## 2 Common Atomic Functions

**Purpose** This chapter describes the atomic functions and events that are common to all PPL components. AF 1- AF 50 are common to all PPL components. See the relevant chapter for protocol-specific \*.exl files.

**Atomic Functions**

<b>AF Number</b>	<b>1</b>
<b>Name</b>	Create New State Machine Context
<b>Description</b>	Creates a new state machine context for the protocol being transferred into. This function must be invoked prior to AF_002 (Jump To State Machine). The Protocol ID can be the currently active protocol or a non-active protocol.
<b>Arg1</b>	<Protocol ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>2</b>
<b>Name</b>	Jump to State Machine
<b>Description</b>	Transfers from the current state machine to the internal state machine indicated by Arg1, which becomes the current state, and the Internal PPL Event indicated by Arg2 is processed.
<b>Arg1</b>	<Internal State ID>
<b>Arg2</b>	<Internal PPL Event>

<b>AF Number</b>	<b>3</b>
<b>Name</b>	Return to State Machine
<b>Description</b>	Returns to the calling PPL state machine. Both AF_001 (Create New State Machine Context) and AF_002 (Jump To State Machine) must be invoked prior to AF_003. The current state machine context will be freed up and the calling state machine's context will be restored.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>11</b>
<b>Name</b>	Generates a PPL Internal Event (Indicated by Config Byte)
<b>Description</b>	Generates a PPL Event indicating the value of the configuration byte indicated by Arg1.
<b>Arg1</b>	<Configuration Byte Offset> 1-200
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>12</b>
<b>Name</b>	Compare PPL Configuration Byte To Value
<b>Description</b>	Tests the value of the configuration byte indicated by Arg1 against the value indicated by Arg2.
<b>Arg1</b>	<Configuration Byte Offset> 1-200
<b>Arg2</b>	<Value>

<b>Test Result</b>	<b>PPL Internal Event</b>
<	0
>	1
=	2

<b>AF Number</b>	<b>13</b>
<b>Name</b>	Generates a PPL Internal Event (Indicated by GPR)
<b>Description</b>	Generates a PPL Event equal to the value of the configuration byte given by the GPR indicated by Arg1.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>14</b>
<b>Name</b>	Compare PPL Configuration Byte to Value
<b>Description</b>	Compares the value of the configuration byte given by the GPR indicated by Arg1 against the value indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Value>

<b>Result</b>	<b>PPL Internal Event</b>
<	0
>	1
=	2

<b>AF Number</b>	<b>15</b>
<b>Name</b>	Compare PPL Configuration Byte In GPR to Byte Offset Value
<b>Description</b>	Tests the value of the configuration byte given by the GPR indicated by Arg1 against the value of the configuration byte indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Configuration Byte Offset> 1-200

<b>Test Result</b>	<b>PPL Internal Event</b>
<	0
>	1
=	2

<b>AF Number</b>	<b>21</b>
<b>Name</b>	Clear General Purpose Register
<b>Description</b>	Clears the GPR indicated by Arg1.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>22</b>
<b>Name</b>	Clear Range of General Purpose Registers
<b>Description</b>	Clears a range of PPL GPRs beginning with the GPR # indicated by Arg1 and including the number of registers indicated by Arg2.
<b>Arg1</b>	<Beginning GPR #> 1-25
<b>Arg2</b>	# of Registers to Clear>

<b>AF Number</b>	<b>23</b>
<b>Name</b>	Save Value in General Purpose Register
<b>Description</b>	Saves the value indicated by Arg2 in the GPR indicated by Arg1.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Value> 0-255

<b>AF Number</b>	<b>24</b>
<b>Name</b>	Increment General Purpose Register
<b>Description</b>	Increments the GPR indicated by Arg1 by the value indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Increment Value>

<b>AF Number</b>	<b>25</b>
<b>Name</b>	Compare GPR to Value
<b>Description</b>	Compares the value in the GPR indicated by Arg1 against the value indicated by Arg2, and returns a decision event to drive the PPL state machine out of an internal decision state.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Value>

<b>Test Result</b>	<b>PPL Internal Event</b>
=	1
other	0

<b>AF Number</b>	<b>26</b>
<b>Name</b>	Compare PPL General Purpose Registers
<b>Description</b>	Tests the value in the GPR indicated by Arg1 against the value in the GPR indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<GPR #> 1-25

<b>Test Result</b>	<b>PPL Internal Event</b>
<	0
>	1
=	2

<b>AF Number</b>	<b>27</b>
<b>Name</b>	Test Value in GPR to Configuration Byte
<b>Description</b>	Tests the value in the GPR indicated by Arg1 against the value of the configuration byte in the GPR indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<GPR #> 1-25

Test Result	PPL Internal Event
<	0
>	1
=	2

<b>AF Number</b>	<b>28</b>
<b>Name</b>	PPL Test GPR for Any Value
<b>Description</b>	Function which generates a decision event based on the value in the GPR indicated by Arg1.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
GPR # N	N

for example: if GPR = 45, then PPL Internal Event = 45

<b>AF Number</b>	<b>29</b>
<b>Name</b>	Decrement General Purpose Register
<b>Description</b>	Decrements the GPR indicated by Arg1 by the value indicated by Arg2. A GPR cannot be decremented below 0. If the decrement value in Arg2 is greater than the value of the GPR in Arg1, the GPR will be set to 0.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Decrement Value>

<b>AF Number</b>	<b>30</b>
<b>Name</b>	Move GPR to GPR
<b>Description</b>	Moves the value stored in the GPR indicated by Arg1 to the GPR indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<GPR #> 1-25

<b>AF Number</b>	<b>31</b>
<b>Name</b>	PPL General Purpose Register <- General Purpose Register Logical AND Config Bytes
<b>Description</b>	Performs logical AND statement between Arg 1 and Arg 2
<b>Arg1</b>	<Register Number> (1:53)
<b>Arg2</b>	<Config Byte Offset> 1:100

<b>AF Number</b>	<b>32</b>
<b>Name</b>	PPL General Purpose Register Number1 <- General Purpose Register Number1 Logical AND General Purpose Register Number2
<b>Description</b>	Performs logical AND statement between Arg 1 and Arg 2. The result is placed into Arg 1
<b>Arg1</b>	<Register 1 Number> (1:53)
<b>Arg2</b>	<Register 2 Number> (1:53)

<b>AF Number</b>	<b>33</b>
<b>Name</b>	PPL General Purpose Register (High 16) < General Purpose Register (High 16) Logical AND Immediate Value
<b>Description</b>	Perform a logical AND statement between MSB of register stored in Arg 1, and the MSBs of register stored in Arg 2. The result is placed in the MSB of Arg 1.
<b>Arg1</b>	<Register Number> 1:25
<b>Arg2</b>	<Value> 0:65535

<b>AF Number</b>	<b>34</b>
<b>Name</b>	PPL General Purpose Register (Low 16) < General Purpose Register (Low 16) Logical AND Immediate Value
<b>Description</b>	Performs a logical AND statement between LSB of
<b>Arg1</b>	<Register Number> 1:25
<b>Arg2</b>	<Value> 0 - 65535

<b>AF Number</b>	<b>35</b>
<b>Name</b>	PPL General Purpose Register <- General Purpose Register Logical OR Config Bytes
<b>Description</b>	Performs logical OR statement between Arg 1 and Arg 2
<b>Arg1</b>	<Register Number> 1:53
<b>Arg2</b>	<Config Byte Offset> 1:100

<b>AF Number</b>	<b>36</b>
<b>Name</b>	PPL General Purpose Register Number1 <- General Purpose Register Number1 Logical OR General Purpose Register Number2
<b>Description</b>	Performs logical OR statement between Arg 1 and Arg 2. Places result into Arg 1 register.
<b>Arg1</b>	<Register 1 Number> 1:53
<b>Arg2</b>	<Register 2 Number> 1:53

<b>AF Number</b>	<b>37</b>
<b>Name</b>	PPL General Purpose Register (High 16) <- General Purpose Register (High 16) Logical OR Immediate Value
<b>Description</b>	Performs logical OR statement between MSB of register stored in Arg 1 and Arg 2. The result is placed in the MSB of Arg 1 register.
<b>Arg1</b>	<Register Number> 1:25
<b>Arg2</b>	<Value> 0:65535

<b>AF Number</b>	<b>38</b>
<b>Name</b>	PPL General Purpose Register (Low 16) <- General Purpose Register (Low 16) Logical OR Immediate Value
<b>Description</b>	Performs a logical OR statement between the LSB of register in Arg 1 and Arg 2. The result is placed in the LSB of Arg 1.
<b>Arg1</b>	<Register Number> 1:25
<b>Arg2</b>	<Value> 0:65535

<b>AF Number</b>	<b>39</b>
<b>Name</b>	PPL Shift Right Logical Gen. Purpose Register
<b>Description</b>	Performs a bit shift right, on the register indicated by Arg 1. The number of bits to shift is indicated by Arg 2.
<b>Arg1</b>	<Register #> 1:53
<b>Arg2</b>	<# bits> 1:64

<b>AF Number</b>	<b>40</b>
<b>Name</b>	Block Event
<b>Description</b>	Blocks an event for later processing in the next normal state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>41</b>
<b>Name</b>	PPL Shift Left Logical GPR
<b>Description</b>	Shifts the value in the GPR indicated by argument 1 left by the number of bits indicated by argument 2.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<# of Bits>

<b>AF Number</b>	<b>42</b>
<b>Name</b>	PPL Load GPR with Contents of 4 or 8 Config Bytes
<b>Description</b>	Loads the value of the PPL Config Bytes beginning with the byte indicated by argument 2 into the GPR indicated by argument 1.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Starting Config Byte>

<b>AF Number</b>	<b>43</b>
<b>Name</b>	PPL Load GPR with Contents of Config Byte
<b>Description</b>	Loads the value of the PPL Config Byte indicated by argument 2 into the GPR indicated by argument 1.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>46</b>
<b>Name</b>	Set PPL Timer
<b>Description</b>	<p>Sets the PPL timer indicated by Arg1 to the value of the Timer ID indicated by Arg2, configured with the <i>PPL Timer Configure</i> message.</p> <p>The Timer Number refers to the three internal PPL multi-purpose timers available per channel. The Timer ID refers to the 100 Generic Protocol Timer IDs that can be configured with the <i>PPL Timer Configure</i> message. A PPL Timer's value can be adjusted by changing the Timer ID value with the <i>PPL Timer Configure</i> message, without modifying the PPL State/Event or Primitive tables.</p>
<b>Arg1</b>	<Timer Number> 1-3
<b>Arg2</b>	<Timer ID> 1-100

The expiration of PPL timers results in the generation of the following PPL events:

<b>Expiration of:</b>	<b>PPL Internal Event</b>
Timer 1	PPLevTIMER1
Timer 2	PPLevTIMER2
Timer 3	PPLevTIMER3

<b>AF Number</b>	<b>47</b>
<b>Name</b>	Disable PPL Timer
<b>Description</b>	Disables the timer indicated by Arg1.
<b>Arg1</b>	<Timer Number> 1-3
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>48</b>
<b>Name</b>	Set PPL Timer in 10ms Units
<b>Description</b>	Activates the PPL timer indicated by Arg1 with the value indicated by Arg2 in 10ms units.
<b>Arg1</b>	<Timer Number> 1-3
<b>Arg2</b>	<Timer Value>

<b>AF Number</b>	<b>49</b>
<b>Name</b>	Set PPL Timer in 100ms Units
<b>Description</b>	Activates the PPL timer indicated by Arg1 with the value indicated by Arg2 in 100ms units.
<b>Arg1</b>	<Timer Number> 1-3
<b>Arg2</b>	<Timer Value>

<b>AF Number</b>	<b>50</b>
<b>Name</b>	Set PPL Timer in 1000ms Units
<b>Description</b>	Activates the PPL timer indicated by Arg1 with the value indicated by Arg2 in 1000ms units.
<b>Arg1</b>	<Timer Number> 1-3
<b>Arg2</b>	<Timer Value>

**EVENTS** The following events are common to all PPL components.

191	PPLevTIMER 1 PPL Timer 1 has expired.
192	PPLevTIMER 2 PPL Timer 2 has expired.
193	PPLevTIMER 3 PPL Timer 3 has expired.
Events 200 - 455	
Internal Event N generated by a test AF to drive the state machine out of a decision type state, where:	
Event N = Event # - 200.	
For example:	
EVENT 200 = Internal Event 0	
EVENT 321 = Internal Event 121	
EVENT 455 = Internal Event 255	
Events 500 - 755	
Internal Event N from Layer 5 generated by a <i>PPL Event Request</i> message, where:	
Event N = Event # - 500.	

### Downloading and Assigning a Protocol

Download the protocol to the switch. See the *Developer's Guides* for downloading methods. Assign the protocol to channels or other objects (such as SS7 links) using the *PPL Assign* message.

Perform any required PPL configuration such as:

- Timers (*PPL Timer Configure*)
- Config Bytes (*PPL Configure*)
- Signaling (*PPL Transmit Signal Configure*, E1 only).

Bring the channels or other objects in service using the *Service State Configure* message.



# 3      T1 Atomic Functions

**Purpose**      This section includes the atomic functions and events associated with the T1 PPL component (0x0003). The PPL state machine, \*.ppl, for each protocol is available on CD ROM.

**Atomic Functions**

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Send L4 a Q.931 Alerting
<b>Description</b>	l3ppl_af_59() sends a Q.931 L3 - L4 Alerting message.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>72</b>
<b>Name</b>	L3PPL Send L4 a Q.931 Alerting with Cut Thru  *Changed from L3PPL Send L4 a Q.931 Alerting.
<b>Description</b>	Sends a Q.931 L3 - L4 Alerting with Cut Thru message.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>74</b>
<b>Name</b>	L3PPL Send L4 a Setup Indication
<b>Description</b>	Sends a Setup Indication to Layer 4
<b>Input Arg1</b>	<Config Byte Offset>
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>76</b>
<b>Name</b>	L3PPL Send L5 INSZ/OUTZ Ctrl Resp with Status in GPR <GPR #>
<b>Description</b>	Sends a INSEIZE_CTRL or OUTSEIZE_CTRL response should an outstanding L5 Inseize/Outseize Control request exist. Response status is stored in the GPR specified in arg1.
<b>Input Arg1</b>	GPR # (1 - 25)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>83</b>
<b>Name</b>	L3PPL Test for Outstanding L5 Msg
<b>Description</b>	Tests for any outstanding L5 messages.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: outstanding L5 messages does not exist. PPLvINT_EVENT_1: outstanding L5 messages exist.

<b>AF Number</b>	<b>118</b>
<b>Name</b>	T1PPLXMIT AB Line Signaling <CFG Byte Offset>
<b>Description</b>	Transmits line signaling which is stored in the CFG Byte offset. The value stored in the CFG Byte offset is expected to contain only AB signaling bit combination.
<b>Input Arg1</b>	Config byte offset that contains line signaling. Range (0 - 3)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>119</b>
<b>Name</b>	L3PPL XMIT ABCD Line Signaling <CFG Byte Offset>
<b>Description</b>	Transmits line signaling which is stored in the CFG Byte offset. The value stored in the CFG Byte offset is expected to contain ABCD signaling bit combination.
<b>Input Arg1</b>	CFG Byte offset that contains line signaling. Range (0 - 15)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>120</b>
<b>Name</b>	T1PPL XMIT Line Signaling <AB bits>
<b>Description</b>	Transmits line signaling which is stored in arg1.
<b>Input Arg1</b>	AB signaling bit combination. (0 - 3)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>177</b>
<b>Name</b>	L3PPL Store Outseize Ctrl Data ICBs
<b>Description</b>	Stores outseize control instructions data ICBs.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>192</b>
<b>Name</b>	L3PPL Test Current Outsize Ctrl Instr Data
<b>Description</b>	Sets up an internal decision event based on the current outsize ctrl instruction data
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0 + outsize ctrl instruction data

<b>AF Number</b>	<b>193</b>
<b>Name</b>	L3PPL Compare Next Outsize Ctrl Instr Addr Sig Type <Addr Sig Type>
<b>Description</b>	Compares the next outsize control instruction data outpulsing address signaling type with the address signaling type passed in arg1 and sets up an internal decision event accordingly.
<b>Input Arg1</b>	Address Signaling Type  01 DTMF 02 MFR1 (host does not include KP or ST) 03 MFR2 04 MFR1 (host includes KP and any ST signal) 05 Dial Pulse 06 MFR1, Transmit Compelled (MF-MDR1)
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0: address signaling type does not equal to arg1. PPLevINT_EVENT_1: address signaling type equal to arg1.

<b>AF Number</b>	<b>212</b>
<b>Name</b>	L3PPL INIT a Channel Purge with Purge Reason in GPR <GPR #>
<b>Description</b>	Initiates a purge passing the purge reason to L4. Purge reason is stored in the GPR.
<b>Input Arg1</b>	<GPR #> 1 - 25
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>226</b>
<b>Name</b>	Send <i>PPL Event Indication</i> Message to the Host with GPR Value
<b>Description</b>	Sends a <i>PPL Event Indication</i> message to the host with the event indicated by Arg1.   The value of the GPR indicated by Arg2 is returned to the host in a PPL GPR Data ICB (Subtype, 0xFF, Data Length = 4).
<b>Arg1:</b>	<PPL Event> See message “PPL Event Indication 0x0043” in the <i>API Reference</i> .
<b>Arg2:</b>	<GPR #> 1-25

<b>AF Number</b>	<b>229</b>
<b>Name</b>	L3 PPL Send L5 Event Indication with General Purpose Register Value <Event ID>
<b>Description</b>	Sends an Event Indication to Layer 5 with GPR value of Event ID
<b>Arg1:</b>	<GPR #> 1-25
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>265</b>
<b>Name</b>	L3PPL Test Previous Outseize Ctrl Instr Data
<b>Description</b>	Tests for previous outseize ctrl instruction data and sets up an internal decision event according.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0 +previous outs ctrl instr data PPLvINT_EVENT_255: previous outs ctrl instr does not exist

<b>AF Number</b>	<b>266</b>
<b>Name</b>	L3PPL Test Previous Outseize Ctrl Instr
<b>Description</b>	Tests for previous outseize ctrl instruction and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0 + previous outs ctrl instr PPLevINT_EVENT_225: previous outs ctrl instr does not exist

<b>AF Number</b>	<b>270</b>
<b>Name</b>	L3PPL Outpulse MFR1 KP Digit using compelled KP Mode
<b>Description</b>	Sends a request to DSP to outpulse digits using compelled KP mode.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>271</b>
<b>Name</b>	L3PPL Continue Outpulse MFR1 Digits using Compelled KP Mode
<b>Description</b>	Sends a request to DSP to remove KP and continue outpulsing digits using compelled KP mode.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>276</b>
<b>Name</b>	PPL Enable Multi Purpose Timer <Timer #> <GPR #>
<b>Description</b>	Enables 1 of 3 multi-purpose timers for use by the PPL. The timer value is stored in GPR indicated in arg2.
<b>Input Arg1</b>	Timer # (1 - 3)
<b>Input Arg2</b>	GPR # (1 - 25)
<b>Outputs</b>	NA

<b>AF Number</b>	<b>279</b>
<b>Name</b>	L3PPL Test CP Event Bit of Onhook <Boolean>
<b>Description</b>	Compares call processing event bit of onhook with the value specified in arg1 and sets up an internal decision event accordingly.
<b>Input Arg1</b>	Boolean (0 -1)
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0: False PPLevINT_EVENT_1: True

<b>AF Number</b>	<b>280</b>
<b>Name</b>	L3PPL Setup Host CP Event of Wink
<b>Description</b>	Sets up a Call Processing event of Wink for possible later reporting to L5. Wink number is extracted from the outsize control message.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>281</b>
<b>Name</b>	L3PPL Setup Host CP Event of Offhook <Boolean>
<b>Description</b>	Sets up a call processing event of offhook for possible later reporting to L5.
<b>Input Arg1</b>	Boolean (0 - 1)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>282</b>
<b>Name</b>	L3PPL Setup Host CP Event of Onhook <Boolean>
<b>Description</b>	Sets up a call processing event of onhook for possible later reporting to L5. Also used for onhook host notify.
<b>Input Arg1</b>	Boolean (0 - 1)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>285</b>
<b>Name</b>	L3PPL Test Outseize Ctrl for any Action ICB
<b>Description</b>	Checks for the outseize ctrl instructions for an Action ICB and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: Null PPLvINT_EVENT_1: Exist

<b>AF Number</b>	<b>286</b>
<b>Name</b>	L3PPL Test Outseize Ctrl for all Data ICB
<b>Description</b>	Checks for the outseize ctrl instruction for data ICB and sets up an internal decision event accordingly. No action ICB is allowed in the outseize control message.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: Null PPLvINT_EVENT_1: Exist

<b>AF Number</b>	<b>287</b>
<b>Name</b>	L3PPL Load Current Outseize Ctrl Instr Data [0], Data [1] into GPR <GPR #>
<b>Description</b>	Loads current outseize ctrl instruction Data [0], Data [1] into GPR.
<b>Input Arg1</b>	GPR # (1 - 25)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>290</b>
<b>Name</b>	T1PPL Test for Flash Timing
<b>Description</b>	Tests if flash timing is enabled.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: Flash timing not enabled PPLvINT_EVENT_1: Flash timing enabled

<b>AF Number</b>	<b>295</b>
<b>Name</b>	T1PPL Outpulse MFR1 Digits
<b>Description</b>	Sends a request to DSP to outpulse MFR1 digits for the stage indicated by the current instruction data.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>300</b>
<b>Name</b>	T1PPL Outpulse DTMF Digits
<b>Description</b>	Sends a request to DSP to outpulse DTMF digits for the stage indicated by the current instruction data.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>305</b>
<b>Name</b>	L3PPL Test Next Inseize Ctrl Instr for Impulsing Digit Str Cnt
<b>Description</b>	Tests the impulsing parameter digit string count for the next inseize ctrl instruction. It is assumed that the next instruction will be receive Stage n.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0: 0 digit strings or invalid string count PPLevINT_EVENT_1: 1 digit string PPLevINT_EVENT_2: 2 digit strings

<b>AF Number</b>	<b>306</b>
<b>Name</b>	L3PPL Test Next Inseize Ctrl Instr for Collection Method
<b>Description</b>	Sets up an internal decision event based upon the next stage's impulsing parameter collection method.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_1: USE_FIXED_NUMBER PPLevINT_EVENT_2: USE_MULTIPLE_TERM_DIGITS PPLevINT_EVENT_3: USE_KP_ST PPLevINT_EVENT_4: USE_FIXED_OR_TERM PPLevINT_EVENT_5: USE_COMPELLED_MODE PPLevINT_EVENT_6: USE_KP_ST_ALL PPLevINT_EVENT_7: USE_COMPELLED_KP PPLevINT_EVENT_20: default

<b>AF Number</b>	<b>310</b>
<b>Name</b>	T1PPL Attach DTMF Digit Receiver
<b>Description</b>	Request to collect DTMF digits. The stage of digits to be collected is determined by the next instruction in the instruction list.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>315</b>
<b>Name</b>	T1PPL Attach MFR1 Digit Receiver
<b>Description</b>	Request to collect MFR1 digits. The stage of digits to be collected is determined by the next instruction in the instruction list.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>317</b>
<b>Name</b>	T1PPL Attach MFR1 Digit Receiver (Get Stage # From Next Instr) <PPL CFG Byte #>
<b>Description</b>	Allocates a MFR1 receiver to collect digits. The stage of digits to be collected is determined by the next instruction in the instruction list. It is like Atomic Function 315 with the added feature of configuring the MFR1 tone reception threshold.
<b>Input Arg1</b>	The first argument is the index into the ppl configuration byte array. The value of this byte is absolute value of the minimum dBm level of the MFR1 digits for detection. Valid entries for this ppl_cfg_byte are between 1 and 30, which equate to -1dBm and -30dBm, respectively. The MFR1 receiver will reject any tones below the detection level.
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>318</b>
<b>Name</b>	T1PPL Attach MFR1 Digit Receiver (Get Stage # From Current Instr) <PPL CFG Byte #>
<b>Description</b>	Allocates a MFR1 receiver to collect digits. The stage of digits to be collected is determined by the current instruction in the instruction list. It is like Atomic Function 316 with the added feature of configuring the MFR1 tone reception threshold.
<b>Input Arg1</b>	The first argument is the index into the ppl configuration byte array. The value of this byte is absolute value of the minimum dBm level of the MFR1 digits for detection. Valid entries for this ppl_cfg_byte are between 1 and 30, which equate to -1dBm and -30dBm, respectively. The MFR1 receiver will reject any tones below the detection level.
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>319</b>
<b>Name</b>	L3PPL Test Inseize Ctrl for any Action ICB
<b>Description</b>	Checks the inseize ctrl instruction for any action ICBs and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: Null PPLvINT_EVENT_1: Exist

<b>AF Number</b>	<b>320</b>
<b>Name</b>	L3PPL Test Current Inseize Ctrl Instr for Addr Signaling Type
<b>Description</b>	Sets up an internal decision event based upon the current stage's inpulsing parameter address signaling type.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_1: DTMF PPLvINT_EVENT_2: MFR1 PPLvINT_EVENT_3: MFR2 PPLvINT_EVENT_4: MFR1_ALL PPLvINT_EVENT_5: DP PPLvINT_EVENT_6: COMPELLED_KP PPLvINT_EVENT_20: Default

<b>AF Number</b>	<b>321</b>
<b>Name</b>	L3PPL Load GPR with Stage Complete Timer for Current Inseize Ctrl Instr <GPR #>
<b>Description</b>	Loads GPR specified in arg1 with the stage complete timer for the current inseize ctrl instruction.
<b>Input Arg1</b>	GPR # (1 - 25)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>322</b>
<b>Name</b>	L3PPL Test Current Inseize Ctrl Instr for Collection Method
<b>Description</b>	Sets up an internal decision event based upon the current stage's inpulsing parameter collection method.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_1: USE_FIXED_NUMBER PPLvINT_EVENT_2: USE_MULTIPLE_TERM_DIGITS PPLvINT_EVENT_3: USE_KP_ST PPLvINT_EVENT_4: USE_FIXED_OR_TERM PPLvINT_EVENT_5: USE_COMPELLED_MODE PPLvINT_EVENT_6: USE_KP_ST_ALL PPLvINT_EVENT_7: USE_COMPELLED_KP PPLvINT_EVENT_20: default

<b>AF Number</b>	<b>323</b>
<b>Name</b>	L3PPL Test Current Inseize Ctrl Instr for Inpulsing Digit Str Cnt
<b>Description</b>	Tests the inpulsing parameter digit string count for the current inseize ctrl instruction.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0: 0 digit strings or invalid string count PPLvINT_EVENT_1: 1 digit string PPLvINT_EVENT_2: 2 digit strings

<b>AF Number</b>	<b>324</b>
<b>Name</b>	L3PPL Load Current Inseize Ctrl Instr Data[0], Data [1] into GPR <GPR #>
<b>Description</b>	Loads current inseize ctrl instruction Data[0], Data [1] into GPR
<b>Input Arg1</b>	GPR # (1 - 25)
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

<b>AF Number</b>	<b>325</b>
<b>Name</b>	L3PPL Test Previous Inseize Ctrl Instr Data
<b>Description</b>	Test for previous inseize ctrl instruction data and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0 + previous inseize ctrl instr data PPLvINT_EVENT_255: previous inseize ctrl instr does not exist

<b>AF Number</b>	<b>326</b>
<b>Name</b>	L3PPL Test Previous Inseize Ctrl Instr
<b>Description</b>	Tests for previous inseize ctrl instruction and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0 + previous inseize ctrl instr does not exist PPLvINT_EVENT_255: previous inseize ctrl does not exist

<b>AF Number</b>	<b>330</b>
<b>Name</b>	L3PPL Test Wink Number in Host Generate CP Event
<b>Description</b>	Tests the wink number in the host generate call processing event and sets up an internal decision event accordingly.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	PPLvINT_EVENT_0 + Wink Number PPLvINT_EVENT_225: No event to process

<b>AF Number</b>	<b>335</b>
<b>Name</b>	L3PPL Do CPA without Line Signaling

<b>Description</b>	Sends a request to do Call Progress Analysis. The class is taken from the outsize control message.
<b>Input Arg1</b>	NA
<b>Input Arg2</b>	NA
<b>Outputs</b>	NA

**PPL Events**

<b>Event Number</b>	<b>Event Name</b>	<b>Event Description</b>
53	L3PPLevL4GEN_CP_EVENT_WINK	Host sends a generate call processing event of Wink on the specified channel.
54	L3PPLev_HOST_CONNECT	Event received when the channel that was outseized received a “Connect” sent from the host.
63	L3PPLevL4_XMIT_FLASH	Indicates host has sent a Call Processing Event of FLASH on the specified channel.
64	L3PPLevL4_XMIT_ONHOOK	Indicates host has sent a Call Processing Event of ONHOOK on the specified channel.
65	L3PPLevL4_XMIT_OFFHOOK	Indicates host has sent a Call Processing Event of OFFHOOK on the specified channel.
82	L3PPLevDSP_RESULT_FIRST_DIGIT_RCVD	Used for compelled KP. Event received when the first KP digit is received.
100	T1PPLevRCV_LINE_SIG_00	Receive event of AB line signaling. A low B low.
101	T1PPLevRCV_LINE_SIG_01	Receive event of AB line signaling. A low B high.
102	T1PPLevRCV_LINE_SIG_10	Receive event of AB line signaling. A high B low.
103	T1PPLevRCV_LINE_SIG_11	Receive event of AB line signaling. A high B high.

# 4 E1 Atomic Functions

**Purpose** This chapter documents the atomic functions associated with the E1 PPL component (0x0001) that are released for customization using the PPL Composer.

**Atomic Functions** System software version 4.0 or newer is required, unless otherwise noted.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Set GPR to # of Digits Received in Stage
<b>Description</b>	Sets the value in the GPR indicated by Arg1 to the number of digits received for the stage indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Stage #> 1-4

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Store Received Digit in GPR
<b>Description</b>	Stores a single digit received in the GPR indicated by Arg1. If a digit durations reported, it is stored in the GPR indicated by Arg2.
<b>GPR 2:</b>	Stage # of the received digits
<b>GPR 3:</b>	String # of the received digits
<b>GPR 4:</b>	Digit to store (index into digit buffer)
<b>GPR 5:</b>	Value of the GPR where the digit is to be stored (1-25).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Store Digits Received in GPR
<b>Description</b>	Stores a single digit received in the GPR indicated by Arg1. If a digit durations reported, it is stored in the GPR indicated by Arg2.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<GPR #> 1-25

<b>AF Number</b>	<b>54</b>
<b>Name</b>	L3PPL Store digit at fwd xmit ctr in General Purpose Register
<b>Description</b>	Sets the GPR indicated by Arg 2 to the digit indexed by the Stage number, as indicated by Arg 1
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<General Purpose Register> 1:25

<b>AF Number</b>	<b>55</b>
<b>Name</b>	L3PPL Store Digit Received in General Purpose Register
<b>Description</b>	Stores a single digit received in the GPR specified by Arg 1
<b>Arg1</b>	<Register Number> 1:25
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	L3PPL Store General Purpose Register in Stage at FWD Xmt Dig Ctr
<b>Description</b>	Stores the digit indexed by the forward transmit digit ctr indicated by Arg 1 into the GPR indicated by Arg 2
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Register Number> 1:25

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Send L4 Out of Service/Blocked
<b>Description</b>	Sends a L3 to L4 Out of Service message with a status of Blocked in order to bring the channel out of service in the call processing layer (L4). This message should be sent when a remote blocked line signaling state is detected.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send a Busied Out Message to Layer 4
<b>Description</b>	Sends a L3 to L4 Busied Out message after generating local blocked line signaling due to processing a host Busied Out message. This results in updating the call processing layer (L4) for the local busied-out condition.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send a Q.931 CONNECT Message to Layer 4
<b>Description</b>	Sends a Q.931 CONNECT message to layer 4 for notification that answer has been detected.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send a DISCONNECT message to Layer 4
<b>Description</b>	Sends a DISCONNECT message to Layer 4 upon the detection of release signaling from the network.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send a Q.931 CLEAR ACK to Layer 4
<b>Description</b>	Sends a Layer 3 to Layer 4 CLEAR ACK upon detection of idle signaling from the network after release. This will result in Layer 4 sending a <i>Channel Released</i> message to the host, provided that Layer 4 has previously sent an L4 Clear Req PPL event (#58) to the state machine.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send an In Service Message to Layer 4
<b>Description</b>	Sends a Layer 3 to Layer 4 In Service message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send an Out of Service (No Reason) Message to Layer 4
<b>Description</b>	Sends Layer 4 an Out of Service (No Reason) message due to distant end failure to release call.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>67</b>
<b>Name</b>	Send Layer 4 a <i>Call Processing Event</i> message from Call Control Instruction
<b>Description</b>	Sends the host a <i>Call Processing Event</i> message with the event indicated by the current call control instruction and any associated data.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Send Layer 5 a <i>Call Processing Event</i> Message
<b>Description</b>	Sends the host a <i>Call Processing Event</i> message with the event indicated by Arg1 and any associated data.
<b>Arg1</b>	<Call Processing Event> 1–17
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Send a Q.931 SETUP INDICATION to Layer 4 with Address Data
<b>Description</b>	Sends a SETUP INDICATION message to Layer 4 for notification of an incoming call, including any address digits collected to this point. This message will result in Layer 4 sending a <i>Request for Service with Address Data</i> message to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Send a Q.931 SETUP INDICATION to Layer 4
<b>Description</b>	Sends a SETUP INDICATION message to Layer 4 for notification of an incoming call. This message will result in Layer 4 sending a <i>Request for Service</i> message to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send an <i>Access Denied</i> Message to Layer 4
<b>Description</b>	Sends an ACCESS DENIED message to Layer 4 for notification of an outseize attempt failure, including the failure reason indicated by Arg1 (this should be one of the Outseize Failure message response status).
<b>Arg1</b>	Arg1: <Outseize Failure Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send a Q.931 ALERTING Message to Layer 4
<b>Description</b>	Sends a Q.931 ALERTING message to Layer 4 for notification of a successful outseizure.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send Layer 4 a Flash Event Notification
<b>Description</b>	Send Layer 4 a Flash Event Notification. Minimum Software Version: 4.1. Sends a Flash Detected message to L4. To enable host notification, see message “Flash Timing Configure 0x0016” in the <i>API Reference</i> .
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>74</b>
<b>Name</b>	L3PPL Send a Setup Indication to Layer 4
<b>Description</b>	Sends a “setup indication” to Layer 4, to report an incoming call with associated address/information signals. The address signaling type is specified by the offset of the Config Byte.
<b>Arg1</b>	<Config Byte Offset> 1 - 200
<b>Arg2</b>	

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Send Layer 4 a <i>Call Processing Event/Digits</i> Message
<b>Description</b>	Sends Layer 4 a <i>Call Processing Event/Digits</i> message, including the stages of digits to be reported indicated by Arg1. This will result in Layer 4 sending a <i>Call Processing Event/Digits</i> message to the host.
<b>Arg1</b>	<Stages of Digits to Report>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Send <i>Call Processing Event /Digits</i> Message
<b>Description</b>	Sends a <i>Call Processing Event /Digits</i> message to the host reporting the digit strings indicated by Arg1.
<b>Arg1</b>	<Stage # Bit Mask>  Bit 0   Stage 1  Bit 1   Stage 2  Bit 2   Stage 3  Bit 3   Stage 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send <i>Call Processing Event/Digit With Timing Information</i> Message
<b>Description</b>	Sends a <i>Call Processing Event/Digit With Timing Information</i> message to the host reporting a single digit. If digit duration reporting is not enabled, the duration will be 0.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send <i>Inseize Control</i> or <i>Outseize Control</i> Response to the Host
<b>Description</b>	Sends an <i>Inseize Control</i> or <i>Outseize Control</i> Response to the host with the error indicated by Arg1. The message sent is determined by which message was originally received.
<b>Arg1</b>	<Response Status> See “API Reference” for values
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send a <i>Call Processing Event</i> Message with Event ID
<b>Description</b>	Sends a <i>Call Processing Event</i> message with the PPL Event indicated by Arg1.
<b>Arg1</b>	<PPL Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Set Receive Stage Status (Minimum Software Version: 4.1)
<b>Description</b>	Updates the receive stage status field for the stage indicated by Arg1 to the value indicated by Arg2.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Value>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Outputpulse BWD R2 Signal in Compelled or Pulsed Mode
<b>Description</b>	Outputpulses a BWD R2 signal in either compelled or pulsed mode, depending on the current FWD R2 signaling state. If a FWD R2 signal is present, the BWD R2 signal will be compelled, otherwise it will be pulsed.
<b>Arg1</b>	<BWD R2 Signal> 1 -15 (for both group A and group B).
<b>Arg2</b>	<Value>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Outputpulse BWD R2 Signal in Compelled Mode
<b>Description</b>	Outputpulses a BWD R2 signal in compelled mode.
<b>Arg1</b>	<BWD R2 Signal> 1 -15 (for both Group A and Group B).
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Outputpulse BWD Pulsed R2 Signal
<b>Description</b>	Outputpulses a BWD R2 signal in pulsed mode. The R2 signal value is indicated in Data Byte 2 of a Generate Call Processing Event instruction.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Outputpulse BWD Compelled R2 Signal
<b>Description</b>	Outputpulses a backward compelled R2 signal. The R2 signal value is indicated in Data Byte 2 of a Generate Call Processing Event instruction.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Outputpulse BWD R2 Signal in Compelled Mode/Generate Completion Event
<b>Description</b>	Outputpulses a backward R2 signal in compelled mode and instructs the MFDSP card to generate PPL Event 46 (R2 COMP CYCLE COMPLETE) when the R2 cycle is complete (silence is detected in both the forward and backward directions). The R2 signal value is indicated in Data Byte 2 of a Generate Call Processing Event instruction. The PPL must wait in a normal state for this event after executing this atomic function to guarantee that the R2 cycle has completed. Once this event is received, MFC R2 reception/transmission can be cancelled using AF 147.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Outputpulse BWD R2 Signal in Compelled Mode/Generate Completion (Using GPR)
<b>Description</b>	<p>Outputpulses a backward R2 signal (from GPR indicated by Arg1) in compelled mode and instructs the MFDSP card to generate PPL Event 46 (R2 COMP CYCLE COMPLETE) when the R2 cycle is complete (silence is detected in both the forward and backward directions).</p> <p>The PPL must wait in a normal state for this event after executing this atomic function to guarantee that the R2 cycle has completed. Once this event is received, MFC R2 reception/transmission can be cancelled using AF 147.</p>
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Outputpulse FWD R2 Signal in Compelled Mode
<b>Description</b>	Outputpulses a forward R2 signal in compelled mode.
<b>Arg1</b>	<FWD R2 Signal> 1-15 (for both Group I and Group II).
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Outputpulse BWD R2 Signal in Compelled or Pulsed Mode
<b>Description</b>	Outputpulses a BWD R2 Signal in either compelled or pulsed mode, depending on the current tone receive state. The R2 signal value is indicated in Data Byte 2 of a <i>Generate Call Processing Event</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Output Next FWD Compelled R2 Signal
<b>Description</b>	Transmits the next FWD compelled R2 signal indicated by the FWD R2 Transmit Digit Counter from the stage indicated by Arg1.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Output BWD R2 Signal in Forced Pulsed Mode
<b>Description</b>	Output a backward R2 signal in pulsed mode and waits until the R2 signal has been pulsed to process the next atomic function.
<b>Arg1</b>	<BWD R2 Signal> 1-15 (for both Group A and Group B).
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	L3PPL Output Single MFR1 Digit
<b>Description</b>	Sends a request to TC to output individual MFR1 digits.
<b>Arg1</b>	<Register Number>
<b>Arg2</b>	<Outp Event Flag>

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Output MFR1 Digits
<b>Description</b>	Sends an internal message to output the MFR1 digit string(s) contained in the Output Stage N Address Data ICB for the stage indicated in Arg1. Note: There must be a least one DSP configured for MFR1 tone transmission for this function to work.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Outpulse MFR1 Digits for Stage Indicated in Outseize Control Instruction
<b>Description</b>	Sends an internal message to output the MFR1 digit string(s) contained in the Outpulse Stage N Address Data ICB for the stage indicated. Note: There must be at least one DSP configured for MFR1 tone transmission for this function to work.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Outpulse DTMF Digits
<b>Description</b>	Sends an internal message to output the DTMF digit string(s) contained in the Outpulse Stage N Address Data ICB for the stage indicated in Arg1.  Note: There must be a least one DSP configured for DTMF tone transmission for this function to work.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Outpulse DTMF Digits for Stage Indicated in Outseize Control Instruction
<b>Description</b>	Sends an internal message to output the DTMF digit string(s) contained in the Outpulse Stage N Address Data ICB for the stage indicated.  Note: There must be a least one DSP configured for DTMF tone transmission for this function to work.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Cancel Outpulsing Digits
<b>Description</b>	Cancels an outstanding outpulse digits request
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Outpulse DTMF Digits
<b>Description</b>	<p>Minimum Software Version: 4.1</p> <p>Initiates the collection of DTMF digits for the stage indicated by Arg1 using the timer values specified by the Generic Timer ID indicated by Arg2.</p>
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Timer ID> 1-100

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Outpulse MFR1 Digits
<b>Description</b>	<p>Minimum Software Version: 4.1</p> <p>Initiates the collection of MFR1 digits for the stage indicated by Arg1 using the timer values specified by the Generic Timer ID indicated by Arg2.</p>
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Timer ID> 1-100

<b>AF Number</b>	<b>105</b>
<b>Name</b>	E1PPL Outpulse MFR1 digits in IFB mode
<b>Description</b>	Sends a request to TC to outpulse MFR1 digits for the stage specified in Arg 1, using IFB mode. No KP/ST Framing
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Number of Cycles> 1:10

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Outpulse Dial Pulse Digits from Outseize Control Instruction
<b>Description</b>	Sends an internal message to outpulse the Dial Pulse digit string(s) contained in the Outpulse Stage N Address Data ICB for the stage indicated in the ICB. Note: A DSP does not need to be configured for this function to work.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Outpulse Dial Pulse Digits from Outseize Control Instruction for Stage Indicated
<b>Description</b>	Sends an internal message to outpulse the Dial Pulse digit string(s) contained in the Outpulse Stage N Address Data ICB for the stage indicated by Arg1. Note: A DSP does not need to be configured for this function to work.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Cancel Outpulse Dial Pulse Digits
<b>Description</b>	Cancels an outstanding request for outpulsing Dial Pulse digits.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	L3PPL Outpulse DP from fwd xmit ctr in General Purpose Register
<b>Description</b>	Sends a request to xsig to outpulse DP digits for stage specified in Arg 1, beginning at the digit specified by the forward xmit digit ctr.
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Outp Event Flag> 0:1 0 = No outpulse complete event 1 = Send outpulse complete event

<b>AF Number</b>	<b>112</b>
<b>Name</b>	Generate Tone
<b>Description</b>	Initiates the generation of the tone indicated by Arg1, for the number of cycles indicated by Arg2.
<b>Arg1</b>	<Tone ID> See message “Connect Tone Pattern 0x002F” in the API Reference.
<b>Arg2</b>	<# of Cycles>

<b>AF Number</b>	<b>113</b>
<b>Name</b>	Generate Dialtone
<b>Description</b>	Initiates the transmission of North American Standard Dialtone.
<b>Arg1</b>	<Not Used>.
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>114</b>
<b>Name</b>	Cancel Tone
<b>Description</b>	Cancels the generation of the tone indicated by Arg1.
<b>Arg1</b>	<Not Used>.
<b>Arg2</b>	<Not Used>

AF Number	121																		
Name	Transmit Line Signaling																		
Description	Sets the transmit line signaling for a channel. Note: CCITT recommends that the C bit be low (0) and the D bit be high (1).																		
Arg1	<p>&lt;ABCD Signaling Bits&gt;</p> <p>The ABCD signaling bits are a binary mask representing the line signaling bit combination to be transmitted. The bit order is as follows:</p> <table><tr><td>Bit #</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>Signal Bit</td><td></td><td></td><td></td><td></td><td>A</td><td>B</td><td>C</td><td>D</td></tr></table>	Bit #	7	6	5	4	3	2	1	0	Signal Bit					A	B	C	D
Bit #	7	6	5	4	3	2	1	0											
Signal Bit					A	B	C	D											
Arg2	<Not Used>																		

<b>AF Number</b>	<b>122</b>
<b>Name</b>	Transmit Default Idle Line Signaling
<b>Description</b>	Transmits the preconfigured idle line signaling ABCD bit pattern.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>123</b>
<b>Name</b>	Enable CAS ABCD Bit Line Signaling Scanning
<b>Description</b>	Activates scanning of the Channel Associated Signaling (CAS) ABCD line signaling bits.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>126</b>
<b>Name</b>	E1PPL Attach MFCR2 Receiver (1 - Forward, 2 - Backward) <PPL-CFG-Byte #>
<b>Description</b>	Allocates a MFR2 receiver for Forward/Backward R2 Reception. The first argument determines the direction of the MFR2 signal. The second argument is the location of the configuration byte that contains the setting for the MFR2 tone reception level.
<b>Arg1</b>	MFR2 direction: 1 - Forward, 2 - Backward
<b>Arg2</b>	Index to ppl_cfg_byte which determines if MFR2 tone reception level is set to the high (-22dB) or low (-35dB). The index can be from 1 - 200.

<b>AF Number</b>	<b>127</b>
<b>Name</b>	Setup for FWD Compelled R2 Reception
<b>Description</b>	Allocates a FWD R2 compelled DSP digit receiver. Note: This is a blocking atomic function.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>128</b>
<b>Name</b>	Setup for BWD Compelled R2 Reception
<b>Description</b>	Allocates a BWD R2 compelled DSP digit receiver. Note: This is a blocking atomic function
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Allocate a FWD R2 Digit Buffer for Current Stage
<b>Description</b>	Allocates a forward R2 digit buffer for storing the current stage of address signaling.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Allocate a FWD R2 Digit Buffer
<b>Description</b>	Allocates and initializes a forward R2 signal digit buffer for the stage indicated by Arg1.
<b>Arg1</b>	Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Reset FWD R2 Receive Digit Counter for Current Stage
<b>Description</b>	Sets the FWD R2 digit reception counter to 0 for the current address signaling stage.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Reset FWD R2 Receive Digit Counter
<b>Description</b>	Initializes the FWD R2 receive digit counter to 0 for the stage indicated by Arg1.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>133</b>
<b>Name</b>	Store BWD R2 Signal as Event
<b>Description</b>	Stores the currently received BWD R2 signal for later reporting as a Call Processing Event.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Set BWD R2 Signal Event Bit
<b>Description</b>	Sets the BWD R2 signal event bit used for later reporting as a Call Processing Event.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Test For FWD Receive Digit Counter Value in Range
<b>Description</b>	Minimum Software Version: 4.1. Test atomic function that compares the number of digits collected in Stage 1 to the minimum and maximum number of digits indicated by Arg1 and Arg2 respectively, and returns an Internal PPL Event according to the results.
<b>Arg1</b>	<GPR #> 1-25
<b>Arg2</b>	<GPR #> 1-25

<b>Test Result</b>	<b>PPL Internal Event</b>
< minimum	1
within range	2
> or = to maximum	3

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Clear Stage/FWD Digit Count
<b>Description</b>	Clears the digit string and resets the digit count for the stage indicated by Arg1.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>139</b>
<b>Name</b>	Send DSP Service Request to Collect Individual Digits/Make Matrix Slave (Addr Sig Type)
<b>Description</b>	Sends a "Request Digits" message to sym to collect individual digits after call setup.
<b>Arg1</b>	<Address Signal Type> 1:2
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>140</b>
<b>Name</b>	Setup for DTMF Reception with Inseize Control Instruction
<b>Description</b>	<p>Allocates a DTMF digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by the current Receive Stage N Inseize Control instruction.</p> <p>Note: This is a blocking atomic function.</p>
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>141</b>
<b>Name</b>	Setup for DTMF Reception
<b>Description</b>	Allocates a DTMF digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by Arg1.

<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>142</b>
<b>Name</b>	Setup for MFR1 Reception with Inseize Control Instruction
<b>Description</b>	Allocates an MFR1 digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by the current Receive Stage N Inseize Control instruction. Note: This is a blocking atomic function.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>143</b>
<b>Name</b>	Setup for MFR1 Reception
<b>Description</b>	Allocates a MFR1 digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by Arg1. Note: This is a blocking atomic function.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>144</b>
<b>Name</b>	Setup for Dial Pulse Reception with Inseize Control Instruction
<b>Description</b>	Allocates a Dial Pulse (DP) digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by the current Receive Stage N Inseize Control instruction.  Note: This is a blocking atomic function.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>145</b>
<b>Name</b>	Setup for Dial Pulse Reception
<b>Description</b>	Allocates a Dial Pulse (DP) digit receiver. Digit reception parameters used for controlling digit collection are determined by the inpulsing parameters associated with the stage indicated by Arg1. Note: This is a blocking atomic function.
<b>Arg1</b>	<Stage #> 1 - 4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>146</b>
<b>Name</b>	Send <i>DSP Service Request</i> to Collect Individual Digits
<b>Description</b>	Sends a request to collect single digits after call setup and report digit to the host with a <i>Call Processing Event</i> of “Digits”. The Address Signaling Type is indicated by Arg1.
<b>Arg1</b>	<Address Signaling Type>  1 = DTMF  2 = MFR1  3= MFR2
<b>Arg2</b>	<Not Used>

<b>Note</b>	<p>The following PPL Configuration Bytes must be configured prior to invoking this function:</p> <p>Bytes 196, 197 Digit Timer (MSB, LSB) = The maximum amount of time to wait for the digit. A value of 0xFFFF disables this timer.</p> <p>Byte 198 0 = disabled, 1 = enabled)</p> <p>Bit 0 = Report the digit at its falling edge (when the first digit is released). If the bit is not set the digit will be reported at its rising edge (when the digit is first pressed).</p> <p>Bit 1 - Ignore “#” character as first digit.</p> <p>Bit 2 = Report digit duration (the amount of time the digit has been pressed). Valid only if configuration bit# 0 is 1.</p> <p>Bytes 199, 200 Minimum Receive Digit Duration Timer (MSB, LSB) = The minimum amount of time for a digit to be present before it is detected and validated.</p>
-------------	---

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Cancel R2 Reception
<b>Description</b>	Releases an MFR2 signal receiver. This function can be used to release R2 receivers used for forward and backward R2 reception.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>148</b>
<b>Name</b>	Collect DTMF Digits
<b>Description</b>	Minimum Software Version: 4.1 Initiates the collection of DTMF digits using the timers defined in the Config Bytes (Arg1) with the values specified by the Generic Timer ID indicated by Arg2.
<b>Arg1</b>	<Config Byte #> 1- 200
<b>Arg2</b>	<Timer ID> 1-100

<b>AF Number</b>	<b>149</b>
<b>Name</b>	Collect MFR1 Digits
<b>Description</b>	Minimum Software Version: 4.1  Initiates the collection of MFR1 digits using the timers defined in the Config Bytes (Arg1) with the values specified by the Generic Timer ID indicated by Arg2.
<b>Arg1</b>	<Config Byte #> 1- 200
<b>Arg2</b>	<Timer ID> 1-100

<b>AF Number</b>	<b>150</b>
<b>Name</b>	Send Request To Do Call Progress Analysis For Class 0 (North American)
<b>Description</b>	Sends an internal message to do Call Progress Analysis for Class 0 (North American). Note: There must be a least one DSP configured for call progress analysis for this function to work.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>151</b>
<b>Name</b>	Test Call Progress Analysis Result and Set up Internal Event
<b>Description</b>	Tests the result of Call Progress Analysis.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Answer	80
Error	81
Continuous On	82
Max Silence Timeout	83
Not Determined Timeout	84
Max Determined Timeout	85
Signaling Answer	86
Dialtone	87
Other	0

<b>AF Number</b>	<b>152</b>
<b>Name</b>	Report Call Progress Analysis Result to the Host
<b>Description</b>	Reports the Call Progress Analysis result indicated by Arg1 to the host.
<b>Arg1</b>	<Call Progress Analysis Result>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>153</b>
<b>Name</b>	Report Call Progress Analysis Result from Atomic Function 151 to the Host
<b>Description</b>	Reports the Call Progress Analysis result from Atomic Function 151 to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>154</b>
<b>Name</b>	Send Request To Do Call Progress Analysis (Class From Arg1)
<b>Description</b>	Sends a request to do Call Progress Analysis for the class indicated by Arg1.

<b>Arg1</b>	<Call Progress Analysis Class>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>155</b>
<b>Name</b>	Send Request To Do Call Progress Analysis (Class from Outsize Control Message)
<b>Description</b>	Sends a request to do Call Progress Analysis for the class indicated by an Outsize Control message.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>156</b>
<b>Name</b>	Cancel Call Progress Analysis
<b>Description</b>	Sends a request to cancel Call Progress Analysis.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>157</b>
<b>Name</b>	CPC Detection
<b>Description</b>	Initiates CPC detection.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>158</b>
<b>Name</b>	Cancel CPC Detection
<b>Description</b>	Cancels CPC detection.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>159</b>
<b>Name</b>	Reset FWD R2 Transmit Digit Counter to 1
<b>Description</b>	Sets the generate FWD R2 signal counter to 1 for initiation of FWD R2 signal transmission.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>160</b>
<b>Name</b>	Increment FWD R2 Transmit Digit Counter by 1
<b>Description</b>	Increments the generate FWD R2 signal counter by 1.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>161</b>
<b>Name</b>	Decrement FWD R2 Transmit Digit Counter by 1
<b>Description</b>	Decrements the generate FWD R2 signal counter by 1.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>162</b>
<b>Name</b>	Decrement FWD R2 Transmit Digit Counter by 2
<b>Description</b>	Decrements the generate FWD R2 signal counter by 2.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>163</b>
<b>Name</b>	Decrement FWD R2 Transmit Digit Counter by 3
<b>Description</b>	Decrements the generate FWD R2 signal counter by 3.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>164</b>
<b>Name</b>	Test if Current FWD Digit Count > Stage Digit Count
<b>Description</b>	Tests if the current FWD digit count is greater than the stage digit count for the stage number indicated by Arg1.
<b>Arg1:</b>	Arg1: <Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

### Test Result      PPL Internal Event

Greater than                      1

Other                                0

<b>AF Number</b>	<b>165</b>
<b>Name</b>	Test if Current FWD Digit Count <>== Stage Digit Count
<b>Description</b>	Tests if the current FWD digit count is greater than, less than, or equal to the stage digit count for the stage number indicated by Arg1.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
==	0
<	1
>	2

<b>AF Number</b>	<b>166</b>
<b>Name</b>	Test if FWD Stage Digit Count == 0
<b>Description</b>	Tests if the current FWD digit count for the stage number indicated by Arg1 equals 0.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
== 0	0
Other	1

<b>AF Number</b>	<b>167</b>
<b>Name</b>	Store FWD Transmit Digit Counter Value in GPR
<b>Description</b>	Minimum Software Version: 4.1  Stores the value of the forward digit counter (i.e. number of digits received) for the current call in a GPR.
<b>Arg1:</b>	<GPR #> 1-25
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>168</b>
<b>Name</b>	Store GPR Value in FWD Transmit Digit Counter
<b>Description</b>	Transfers the contents of a General Purpose register into the forward digit counter for the current call.
<b>Arg1:</b>	<GPR #> 1-25

<b>Arg2:</b>	<Not Used>
--------------	------------

<b>AF Number</b>	<b>169</b>
<b>Name</b>	L3PPL Test if Current Xmit Digit Ctr = Stage Digit Count
<b>Description</b>	Checks that the total number of FWD digits to be transmitted is equal to the current FWD transmit digit counter.
<b>Arg1</b>	<Stage Number> 1:25
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>170</b>
<b>Name</b>	Store FWD R2 Signal
<b>Description</b>	Stores the current FWD R2 signal in the digit buffer assigned to the receive digit stage indicated by Arg1. R2 FWD signals are stored in BCD encoded format. The location within the digit buffer where the signal is stored is determined by the current FWD digit counter.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>171</b>
<b>Name</b>	Test If Digit Count == 0xFF
<b>Description</b>	Tests if the digit count for the stage number indicated by Arg1 equals 0xFF.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
== 0xFF)	2
< 0xFF	1
Other	0

<b>AF Number</b>	<b>172</b>
<b>Name</b>	Setup Host <i>Call Processing Event/Digits</i>
<b>Description</b>	Sets up a Call Processing Event/Digits message to send to the host upon collection of a complete stage of digits.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>173</b>
<b>Name</b>	Increment Receive FWD R2 Digit Counter
<b>Description</b>	Increments the current receive stage digit counter due to reception of a FWD R2 signal.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>174</b>
<b>Name</b>	Store L4 Inseize/Outseize ICBsr
<b>Description</b>	Stores Layer 4 Inseize or Outseize Control ICBs for later processing. These ICBs contain inseize or outseize instructions and related data (i.e.: R2 signal to generate). Layer 4 will send Layer 3 an <i>Inseize Control</i> or <i>Outseize Control</i> message after receiving and preprocessing a host <i>Inseize Control</i> or <i>Outseize Control</i> message.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>175</b>
<b>Name</b>	Test for Prior Sending of Layer 3 to Layer 4 SETUP INDICATION
<b>Description</b>	<p>Tests if an L3 to L4 SETUP INDICATION message has been previously sent.</p> <p>Note: A SETUP INDICATION is sent to the Call Processing Layer (L4) to indicate that an incoming call has been received.</p>
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
Sent	1
Not Sent	0

<b>AF Number</b>	<b>179</b>
<b>Name</b>	Use Pre-programmed Inseize Instruction List
<b>Description</b>	Points a channel to the inseize instruction list preconfigured on the channel with the <i>Inseize Instruction List Configure</i> message.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>180</b>
<b>Name</b>	Use Instructions in <i>Inseize Control</i> or <i>Outseize Control</i> message
<b>Description</b>	<p>Points a channel to the inseize or outseize instruction list provided by the host through an interactive <i>Inseize Control</i> or <i>Outseize Control</i> message.</p> <p>Note: This function can be used for both inseize and outseize instruction processing.</p>
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>181</b>
<b>Name</b>	Reset Instruction Index
<b>Description</b>	Resets the current inseize or outseize instruction pointer to instruction 1. Note: This function can be used for both inseize and outseize instruction processing.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>182</b>
<b>Name</b>	Increment Instruction Index
<b>Description</b>	Increments the current inseize or outseize instruction pointer to the next instruction #. Note: This function can be used for both inseize and outseize instruction #. processing
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>183</b>
<b>Name</b>	Test Current Inseize Control Instruction
<b>Description</b>	Tests for the current Inseize Control instruction
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
Null	0
Report Call Processing Event	1
Generate Call Processing Event	2
Receive Stage n Data	3
Wait for Host Control	4
Report Incoming Call	5
Report Incoming Call with Data	6
Generate Inseize ACK	7
Send Host Inseize ACK	8
Use Instruction List	9
Other	20

<b>AF Number</b>	<b>184</b>
<b>Name</b>	Set Call Control Instruction Index to Host Value
<b>Description</b>	Sets the call control instruction index to that indicated in the Use Instruction List ICB sent in an interactive <i>Inseize Control</i> or <i>Outseize Control</i> message by the host.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>185</b>
<b>Name</b>	3
<b>Description</b>	Tests the stage number in the current Inseize Control Instruction. The stage # is determined by the Receive Stage N Address Data <i>Inseize Control</i> instruction currently being processed.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
Stage # 1	1
Stage # 2	2
Stage # 3	3
Stage # 4	4
Other	0xFF

<b>AF Number</b>	<b>186</b>
<b>Name</b>	Test Next Inseize Control Instruction
<b>Description</b>	Tests if the next Inseize Control Instruction is Receive Stage N Address Data.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
Receive Stage N Address Data	1
Other	0

<b>AF Number</b>	<b>187</b>
<b>Name</b>	Test Inseize Control <i>Generate Call Processing Event</i> Instruction
<b>Description</b>	Tests if the next Inseize Control Instruction is Receive Stage N Address Data.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Result	PPL Internal Event
ANI Request Off Hook	5

Result	PPL Internal Event
Wink 1	6
Wink 2	7
Wink 3	8
Wink 4	9
Wink 5	10
Wink 6	11
Wink 7	12
Wink 8	13
BWD Pulsed R2 Signal	1
BWD Compelled R2 Signal	2
BWD Compelled R2 Signal with Cycle Completion	3
BWD Compelled or Pulsed R2 Signal	4

<b>AF Number</b>	<b>188</b>
<b>Name</b>	Test Next Inseize Control Instruction for Address Signaling Type
<b>Description</b>	Tests for the receive signaling type from the inpulsing parameters of a stage. The stage number is obtained from the next Inseize Control Instruction, as indicated by the current instruction index (provided that it is a Receive Stage N instruction)..
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
DTMF	1
MFR1	2
MFR2	3
Other	20

<b>AF Number</b>	<b>189</b>
<b>Name</b>	Test Address Signaling Type of Stage N Inpulsing Parameters
<b>Description</b>	Tests the inpulsing parameters of the stage indicated by Arg1 for its receive signaling type.
<b>Arg1:</b>	Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
DTMF	1
MFR1	2
MFR2	3
Other	20

<b>AF Number</b>	<b>190</b>
<b>Name</b>	Return Inpulsing Signaling Type
<b>Description</b>	Tests for the receive signaling type from the inpulsing parameters of a stage. The stage number is obtained from the current Inseize Control Instruction, as indicated by the current instruction index (provided that it is a Receive Stage N instruction).
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Test Result	PPL Internal Event
DTMF	1
MFR1	2
MFR2	3
Other	20

<b>AF Number</b>	<b>194</b>
<b>Name</b>	Test for Outseize Control ICB Buffer
<b>Description</b>	Tests if the Outseize Control ICB buffer is attached to the Layer 4 to Layer 3 CALL REQUEST message.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>Result</b>	<b>PPL Internal Event</b>
Yes	1
No	0

<b>AF Number</b>	<b>195</b>
<b>Name</b>	Test for Host Outseize Control Instructions
<b>Description</b>	Tests if there are Outseize Control Instructions from the host.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>Result</b>	<b>PPL Internal Event</b>
Yes	1
No	0

<b>AF Number</b>	<b>196</b>
<b>Name</b>	Use Pre-programmed Outseize Instruction List
<b>Description</b>	Points a channel to use the channel's pre-programmed outseize instruction list.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>197</b>
<b>Name</b>	Test Current Outsize Control Instruction
<b>Description</b>	Tests for the current Outsize Control instruction.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Null	0
Scan for Wink	1
Scan for ANI Request Off-hook	2
Scan for Dialtone	3
Report Call Processing Event	4
Outpulse Stage N Digits	5
Wait for Host Address Data	6
Wait for Host Control	7
Send Host Acknowledgment	8
Do Call Progress Analysis	9
Seize	10
Use Pre-programmed Instruction List	11
Cancel R2 Receiver	13
Scan for Backward R2 Signal	14
Wait for Host Control with Answer Supervision	15
Other	20

<b>AF Number</b>	<b>198</b>
<b>Name</b>	Test for Outpulsing Data ICB
<b>Description</b>	Tests if there are host-supplied outpulsing digits for the current stage.
<b>Arg1:</b>	<Not Used>

<b>Arg2:</b>	<Not Used>
--------------	------------

Test Result	PPL Internal Event
Yes	1
No	0

<b>AF Number</b>	<b>199</b>
<b>Name</b>	Test Address Signaling Type of Outpulse Stage N Address Data for Current Stage
<b>Description</b>	Tests the address signaling type in the Outpulse Stage N Address Data ICB for the current stage. The stage number is obtained from the current outsize control instruction from list, provided that it is an Outpulse Stage N instruction.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

Result	PPL Internal Event
DTMF	1
MFR1	2
MFR2	3
MFR1 ALL	4
Dial Pulse	5
Compelled KP	6
Other	20

<b>AF Number</b>	<b>200</b>
<b>Name</b>	Transfer Remaining Digits
<b>Description</b>	Takes digits remaining from the stage indicated by Arg1 and dynamically creates a Data ICB with the remaining address data for the stage indicated by Arg2.
<b>Arg1:</b>	Stage #> 1 - 4
<b>Arg2:</b>	<Stage #> 1 - 4  Note: If a Data ICB already exists for the stage indicated by Arg2, any digits it contains will be over-written by the digits from the stage indicated by Arg1.

<b>AF Number</b>	<b>201</b>
<b>Name</b>	Return Config
<b>Description</b>	Byte #  Stores the Config Offset where Outseize Control Instructions end in the GPR indicated by Arg1. The Config Byte indicated by Arg2 indicates where the Outseize Control Instructions begin.
<b>Arg1:</b>	Arg1: <GPR #> 1-25
<b>Arg2:</b>	<Config Byte #> 1- 200

<b>AF Number</b>	<b>202</b>
<b>Name</b>	Test for Outpulsing Data ICB
<b>Description</b>	Minimum Software Version: 4.1. Tests if there are host-supplied outpulsing digits for the current stage
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>Result</b>	<b>PPL Internal Event</b>
No	0
Yes	1

<b>AF Number</b>	<b>203</b>
<b>Name</b>	Test Address Signaling Type of Outpulse Stage N Address Data for Current Stage
<b>Description</b>	Minimum Software Version: 4.1. Tests the address signaling type in the Outpulse Stage N Address Data ICB for the stage indicated by Arg1.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
DTMF	1
MFR1	2
MFR2	3
MFR1 ALL	4
Dial Pulse	5
Compelled KP	6
Other	20

<b>AF Number</b>	<b>204</b>
<b>Name</b>	Store Digit Collection Status
<b>Description</b>	Stores the digit collection status for the stage indicated in the current Receive Stage N Address Data inseize control instruction.
<b>Arg1:</b>	<Status #>  66 - Digits Received  67 - First Digit Timeout  68 - Inter-digit Timeout  69 - Digits Complete Timeout
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>205</b>
<b>Name</b>	Store Digits
<b>Description</b>	Stores a DSP decoded digit string in the inpulsing stage digit buffer associated with the stage indicated in the current Receive Stage N Address Data inseize control instruction data.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>206</b>
<b>Name</b>	Store Digits
<b>Description</b>	Stores a DSP decoded digit string in the inpulsing stage digit buffer associated with the stage indicated by Arg1.
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>207</b>
<b>Name</b>	Update Timeout Status
<b>Description</b>	<p>Minimum Software Version: 4.1</p> <p>Updates the Receive digit status for the stage indicated by Arg2 according to the test results shown below. The status is reported in the <i>Request for Service with Data</i> message.</p>
<b>Arg1:</b>	<Timeout Value>
<b>Arg2:</b>	<Stage #> 1 - 4

<b>Test Result</b>	<b>Status</b>
Digits Received	0x10 (Positive ACK)
First Digit Timeout	0x81 (Permanent Signal Condition)
Interdigit Timeout	0x80 (Partial Dial Condition)
Digits Complete Timeout	0x92 (Impulsing Complete Timeout)

<b>AF Number</b>	<b>208</b>
<b>Name</b>	Store Single Digit in Stage
<b>Description</b>	<p>Minimum Software Version: 4.1</p> <p>Stores a single digit in the Receive Stage digit buffer for the stage indicated by Arg1. PPL Event 66 (DSP Result Digits) must be the event received immediately preceding this AF.</p>
<b>Arg1:</b>	<Stage #> 1 - 4
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>210</b>
<b>Name</b>	Initiate a Channel Purge
<b>Description</b>	Resets a channel due to an unrecoverable error. Any associations which exist with this channel will also get implicated in the purging process.

<b>Arg1:</b>	<Purge Reason>  Refer to message “DSO Status Change 0x0042” in the <i>API Reference</i> for purge reasons.
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>211</b>
<b>Name</b>	Ignore PPL Event
<b>Description</b>	Event will be ignored (will not result in the generation of an Invalid Event indication)..
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>215</b>
<b>Name</b>	Change PPL Protocol
<b>Description</b>	Changes a channels protocol to the protocol indicated by Arg1.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>216</b>
<b>Name</b>	Restore Channel’s Preconfigured PPL Protocol
<b>Description</b>	Restores a channel's PPL protocol to the default preassigned protocol.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>225</b>
<b>Name</b>	Send <i>PPL Event Indication</i> Message to the Host
<b>Description</b>	Sends a <i>PPL Event Indication</i> message to the host with the event indicated by Arg1. The ICB Count is 0.
<b>Arg1:</b>	<PPL Event> See message “PPL Event Indication 0x0043” in the <i>API Reference</i> .

<b>Arg2:</b>	<Not Used>
--------------	------------

<b>AF Number</b>	<b>226</b>
<b>Name</b>	Send <i>PPL Event Indication</i> Message to the Host with GPR Value
<b>Description</b>	Sends a <i>PPL Event Indication</i> message to the host with the event indicated by Arg1.   The value of the GPR indicated by Arg2 is returned to the host in a PPL GPR Data ICB (Subtype, 0xFF, Data Length = 4).
<b>Arg1:</b>	<PPL Event> See message “PPL Event Indication 0x0043” in the <i>API Reference</i> .
<b>Arg2:</b>	<GPR #> 1-25

<b>AF Number</b>	<b>227</b>
<b>Name</b>	Send <i>PPL Event Request ACK</i> to Host
<b>Description</b>	Sends a response to a <i>PPL Event Request</i> message with the status value indicated by Arg1.
<b>Arg1:</b>	<Value>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>229</b>
<b>Name</b>	L3 PPL Send L5 Event Indication with General Purpose Register Value <Event ID>
<b>Description</b>	Sends an Event Indication to Layer 5 with GPR value of Event ID
<b>Arg1:</b>	<GPR #> 1-25
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>235</b>
<b>Name</b>	Connect Recorded Announcement (Announcement ID)
<b>Description</b>	This function connects a single announcement.
<b>Arg1:</b>	<Announcement ID> 0 to 1023.

<b>Arg2:</b>	<Options>  Bit Mask. Setting the bit to 1 enables the event.  bit 0 = Start Announcement  bit 1 = Terminate Announcement  bit 2 = Barge In
--------------	--

<b>AF Number</b>	<b>236</b>
<b>Name</b>	Connect Recorded Announcement (Config Byte)
<b>Description</b>	This function connects a single announcement. Arg1 indicates the index into the Config Byte containing the Announcement ID. Arg2 indicates the event options.
<b>Arg1:</b>	<Config Byte #> 1- 200
<b>Arg2:</b>	<Options>  Bit Mask. Setting the bit to 1 enables the event.  bit 0 = Start Announcement  bit 1 = Terminate Announcement  bit 2 = Barge In

<b>AF Number</b>	<b>237</b>
<b>Name</b>	Connect Recorded Announcement (GPR)
<b>Description</b>	This function connects a single announcement. Arg1 indicates the GPR containing the Announcement ID. Arg2 indicates the event options.
<b>Arg1:</b>	<GPR #> 1- 25
<b>Arg2:</b>	<Options>  Bit Mask. Setting the bit to 1 enables the event.  bit 0 = Start Announcement  bit 1 = Terminate Announcement  bit 2 = Barge In

<b>AF Number</b>	<b>238</b>
<b>Name</b>	Connect Recorded Announcement (GPR/Config Byte)
<b>Description</b>	This function connects a single announcement. Arg1 indicates the GPR containing the index into the Config Byte containing the Announcement ID. Arg2 indicates the event options.
<b>Arg1:</b>	<GPR #> 1- 25
<b>Arg2:</b>	<Options>  Bit Mask. Setting the bit to 1 enables the event.  bit 0 = Start Announcement  bit 1 = Terminate Announcement  bit 2 = Barge In

<b>AF Number</b>	<b>239</b>
<b>Name</b>	Connect Chained Recorded Announcement (GPR)
<b>Description</b>	<p>This function connects a chained announcement without start and stop events.</p> <p>Arg1 indicates the GPR containing the Announcement ID of the first announcement. Subsequent announcements are stored in subsequent GPRs. Arg2 indicates the GPR containing the number of announcements to be played.</p>
<b>Arg1:</b>	<GPR #> 1- 25
<b>Arg2:</b>	<GPR #> 1- 25

<b>AF Number</b>	<b>240</b>
<b>Name</b>	Connect Chained Recorded Announcement with Start Event (GPR)
<b>Description</b>	<p>This function connects a chained announcement with start event.</p> <p>Arg1 indicates the GPR containing the Announcement ID of the first announcement. Subsequent announcements are stored in subsequent GPRs.</p> <p>Arg2 indicates the GPR containing the number of announcements to be played.</p>
<b>Arg1:</b>	<GPR #> 1- 25
<b>Arg2:</b>	<GPR #> 1- 25

<b>AF Number</b>	<b>241</b>
<b>Name</b>	Connect Chained Recorded Announcement with Termination Event (GPR)
<b>Description</b>	<p>Minimum Software Version: 4.1</p> <p>This function connects a chained announcement with stop event.</p> <p>Arg1 indicates the GPR containing the Announcement ID of the first announcement. Subsequent announcements are stored in subsequent GPRs.</p> <p>Arg2 indicates the GPR containing the number of announcements to be played.</p>
<b>Arg1:</b>	<GPR #> 1- 25
<b>Arg2:</b>	<GPR #> 1- 25

<b>AF Number</b>	<b>243</b>
<b>Name</b>	Cancel Announcement
<b>Description</b>	This functions disconnect a channel from an announcement.
<b>Arg1:</b>	<Not Used>
<b>Arg2:</b>	<Not Used>

<b>AF Number</b>	<b>341</b>
<b>Name</b>	L3PPL OUTP Non-Compelled R2 Sig w/o Blocking
<b>Description</b>	Sends a request to TC to output a non-compelled R2 signal. It does not enter an internal blocking state to wait for the R2 signal to be outputted.
<b>Arg1</b>	<R2 sig> 1:15
<b>Arg2</b>	<direction> 1:2

<b>AF Number</b>	<b>342</b>
<b>Name</b>	3PPL OUTP Next Non-Compelled FWD R2 Sig w/o Blocking
<b>Description</b>	Sends a request to TC to output non-compelled forward R2 signal, following the forward R2 transmit digit counter. Updates the digit counter and enters blocking state until R2 signals are outputted.
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Digit Count> 1:255

<b>AF Number</b>	<b>343</b>
<b>Name</b>	L3PPL OUTP Next Non-Compelled FWD R2 Sig w/o Blocking
<b>Description</b>	Sends a request to TC to output nest non-compelled forward R2 signal, following the forward R2 transmit digit counter. Updates the forward R2 transmit digit counter. Does not enter internal blocking state to wait for R2 signal to be outputted.
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Digit Count> 1:255

<b>AF Number</b>	<b>344</b>
<b>Name</b>	L3PPL Setup Non-Compelled FWD R2 Digit Reception
<b>Description</b>	Sends a request to TC to attach a non-compelled forward R2 signal receiver for non-compelled forward R2 signals of the stage specified in Arg 1. Enters an internal blocking state before proceeding, until the receiver is attached.
<b>Arg1</b>	<Stage Number> 1:4
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>345</b>
<b>Name</b>	L3PPL Setup Non-Compelled R2 Digit Reception Indefinitely
<b>Description</b>	Sends a request to TC to attach a non-compelled R2 signal receiver for non-compelled forward or backward R2 signals. The receiver is attached until a Cancel Digit Reception is sent. Before proceeding, it enters an internal blocking state until the receiver is attached.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>346</b>
<b>Name</b>	L3PPL Setup Non-Compelled R2 Digit Reception for Fixed Digits
<b>Description</b>	Sends a request to TC to attach a non-compelled R2 signal receiver for non-compelled forward or backward R2 signals. The receiver is attached until the specified number of digits is received. before proceeding, it enters an internal blocking state until the receiver is attached.
<b>Arg1</b>	<Digit Count> 1:255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>347</b>
<b>Name</b>	L3PPL OUTP Non-Compelled BWD R2 Sig in Inseize Instruction
<b>Description</b>	Sends a request to TC to outpulse a non-compelled backward R2 signal in pulsed mode. This R2 signal is stored in a Generate Call Processing Event message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>360</b>
<b>Name</b>	E1PPL Attach MFR1 Digit Receiver (Get Stage # From Next Instr) <PPL CFG Byte #>
<b>Description</b>	Allocates a MFR1 receiver to collect digits. The stage of digits to be collected is determined by the current instruction in the instruction list and the MFR1 tone reception threshold is determined by argument 1.
<b>Arg1</b>	The first argument is the index into the ppl configuration byte array. The value of this byte is absolute value of the minimum dBm level of the MFR1 digits for detection. Valid entries for this ppl_cfg_byte are between 1 and 30, which equate to -1dBm and -30dBm, respectively. The MFR1 receiver will reject any tones below the detection level.
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>361</b>
<b>Name</b>	T1PPL Attach MFR1 Digit Receiver (Get Stage # From Current Instr) <PPL CFG Byte #>
<b>Description</b>	Allocates a MFR1 receiver to collect digits. The stage of digits to be collected is determined by the current instruction in the instruction list and the MFR1 tone reception threshold is determined by argument 1.
<b>Arg1</b>	The first argument is the index into the ppl configuration byte array. The value of this byte is absolute value of the minimum dBm level of the MFR1 digits for detection. Valid entries for this ppl_cfg_byte are between 1 and 30, which equate to -1dBm and -30dBm, respectively. The MFR1 receiver will reject any tones below the detection level.
<b>Arg2</b>	<Not Used>

# 5      Layer 4 Atomic Functions

**Purpose**      This chapter includes the atomic functions and events associated with the Layer 4 PPL component.

## Call Control Channel Management (0x0061)

---

**Atomic Functions** The following Atomic Functions (AF) are specific to the CSP Call Control Channel Management component (0x0061).

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Store a L5 message type and sequence # from current message and return L5 reference ID in GPR
<b>Description</b>	Allocates a node in the L5 reference linked list, stores the message type and sequence number, and returns the reference ID in the GPR # specified in Arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	ACK L5 message using L5 outgoing buffer for L5 reference ID
<b>Description</b>	Sends an acknowledgment for the message in the L5 message linked list, de-allocates the entry, and frees the reference ID. Also returns internal event 1 if the reference ID is found and the ACK is sent and 0 on failure.
<b>Arg1</b>	<GPR # containing Layer 5 Reference ID>
<b>Arg2</b>	<GPR # containing ACK Status>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Block Invalid Events for State
<b>Description</b>	Queues invalid events in the next state until the next state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send L5 PPL Event Indication using outgoing L5 buffer
<b>Description</b>	Sends a <i>PPL Event indication</i> message to L5 with data in the outgoing L5 buffer.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Clear L5 Reference ID
<b>Description</b>	De-allocates the L5 reference link with the specified ID.
<b>Arg1</b>	<GPR # containing the Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	ACK L5 message using L5 outgoing buffer for L5 reference ID with more status
<b>Description</b>	Sends an acknowledgment for the message in the L5 message LL. Also de-allocates the entry and frees the Reference ID. The more status is the last normal state of the PPL state machine. Also returns internal event 1 if the reference ID is found and the ACK is sent and 0 on failure.
<b>Arg1</b>	<GPR # containing Reference ID>
<b>Arg2</b>	<ACK Status>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Stop all CH Timers
<b>Description</b>	Cancels any outstanding CH timers
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Purge channel with reason
<b>Description</b>	Initiates a purge of the channel any associated channel through the CM.
<b>Arg1</b>	<Reason>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Test Host message resend enabled flag
<b>Description</b>	Tests the host configuration resend enable flag for messages
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Test Host RFS retry forever flag
<b>Description</b>	Test the host configuration RFS resend flag.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	ACK L5 message using L5 outgoing buffer for L5 Reference ID
<b>Description</b>	Sends an acknowledgment for the message in the L5 message LL. Also de-allocates the entry and frees the Reference ID.
<b>Arg1</b>	<GPR # containing Layer 5 Reference ID>
<b>Arg2</b>	<ACK Status>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Update local channel parameters based on RW L3 data and incoming message in buffer
<b>Description</b>	Updates the shared RW local channel parameters based on the private RW channel parameters and any appropriate TLVs in the incoming buffer specified by arg1. This is how dB Padding and release modes are dynamically set.
<b>Arg1</b>	<Buffer>  0 = Use default (no buffer) 1 = Layer 5 Incoming Buffer 2 = Layer 4 Incoming Buffer
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Test for DSP Resource Free Request in incoming message buffer
<b>Description</b>	Tests for a DSP Resource Free TLV for the channel in the selected buffer. Returns internal event 0 if the TLV is not found, internal event 1 = free digit receiver, internal event 2 = free call progress transmitter, internal event 3 = free both.
<b>Arg1</b>	<Buffer>  1 = Layer 5 Incoming Buffer 2 = Layer 4 Incoming Buffer
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Test Host Link Connection Status
<b>Description</b>	Tests host link connection status.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Result</b>	0 - Not Connected 1 - Connected

<b>AF Number</b>	<b>76</b>
<b>Name</b>	L4CH Test host port availability based on port ID in GPR
<b>Description</b>	Tests the real-time availability of the specified host port.
<b>Arg1</b>	<GPR w/ host port ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>77</b>
<b>Name</b>	L4CH Load host port ID into GPR
<b>Description</b>	Loads the shared RW host port ID into the GPR.
<b>Arg1</b>	<GPR for host port ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	L4CH Set host port ID from GPR
<b>Description</b>	Sets the shared RW host port ID based on the value of the specified GPR.
<b>Arg1</b>	<GPR with host port ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	L4CH Load host port ID from L5 ref ID into GPR
<b>Description</b>	Takes the host port from the L5 information referenced by L5 ref ID and stores the port ID into the GPR.
<b>Arg1</b>	<GPR w/ L5 ref ID>
<b>Arg2</b>	<GPR for host port ID>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Store L5 data in L5 incoming buffer
<b>Description</b>	Transfers any data buffer attached to the L5 message to the L5 incoming buffer. Any data previously in the L5 incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Store L4 data in L4 incoming buffer
<b>Description</b>	Transfers any data buffer attached to the L4 message to the L4 incoming buffer. Any data previously in the L4 incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Store L3 data in L3 incoming buffer
<b>Description</b>	Transfers any data buffer attached to the L3 message to the L3 incoming buffer. Any data previously in the L3 incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Transfer data from L5 incoming to L3 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Transfer data from L5 incoming to L4 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Transfer data from L4 incoming to L3 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Transfer data from L3 incoming to L4 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Transfer data from L3 incoming to L5 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Clear L3 Outgoing Buffer
<b>Description</b>	De-allocates any L3 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Clear L4 Outgoing Buffer
<b>Description</b>	De-allocates any L4 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Clear L5 Outgoing Buffer
<b>Description</b>	De-allocates any L5 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Test for L3 Outgoing Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Test for L4 Outgoing Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Test for L5 Outgoing Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Test for L3 Incoming Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Test for L4 Incoming Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Test for L4 Incoming Buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Clear L3 Incoming Buffer
<b>Description</b>	De-allocates any L3 incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Store TC data in TC incoming buffer
<b>Description</b>	Transfers any data buffer attached to the TC message to the TC incoming buffer. Any data previously in the TC incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Test for TC incoming buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Clear TC incoming buffer
<b>Description</b>	Tests for the presence of the specified buffer. Return through an IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Transfer data from TC incoming to L5 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Transfer data from TC incoming to L4 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Transfer data from TC incoming to L3 outgoing
<b>Description</b>	Transfers data from one buffer to another using the mode specified in Arg1. The move action (0) moves the pointer and clears the source buffer. The copy action (1) makes an actual copy of the buffer and leaves the original in the incoming buffer.
<b>Arg1</b>	<Transfer Mode>  0 = Move 1 = Copy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Clear L4 Incoming Buffer
<b>Description</b>	De-allocates any L4 incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Clear L5 Incoming Buffer
<b>Description</b>	De-allocates any L5 incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Store L5 call processing event from L5 incoming buffer into
<b>Description</b>	Stores the call processing event id into the GPR.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	L4CH transfer Host TLV from/to buffers
<b>Description</b>	L4CH transfer Host TLV from/to working buffer to/from the other buffer
<b>Arg1</b>	<p>&lt;XYZ&gt;</p> <p>X - Action 0 : To Working Buffer 1 : From Working Buffer</p> <p>Y - Buffer Number 0 : Working 1: TC 3: L3 4: L4 5: L5</p> <p>Z - Outgoing/Incoming 0: Outgoing 1: Incoming (when y = 1 then z can be either 0 or 1)</p> <p><b>Note:</b> possible values of arg1: 050, 150, 051, 151, 040, 140, 041, 141, 030, 130, 031, 131, 010 (or 011), 110 (or 111)</p>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	L4CH Buffer Test
<b>Description</b>	Function to test for A Given Buffer.
<b>Arg1</b>	<Buffer> 0 = Working 1 = TC 3 = L3 4 = L4 5 = L5
<b>Arg2</b>	<Outgoing/Incoming> (Ignored when Arg1 = 0 or 1) 0: Outgoing Buffer 1: Incoming Buffer  Return Value: PPLevINT_EVENT_0 - Buffer not present PPLevINT_EVENT_1 - Buffer present

<b>AF Number</b>	<b>110</b>
<b>Name</b>	L4CH Buffer Move/Copy
<b>Description</b>	L4CH Buffer Move/Copy from /To (L5,L4,L3, TC and Working )
<b>Arg1</b>	<Action>  0: Move 1: Copy
<b>Arg2</b>	<Buffer>  0 = Working 1 = TC 3 = L3 4 = L4 5 = L5

<b>AF Number</b>	<b>111</b>
<b>Name</b>	
<b>Description</b>	L4CH Clear Buffer
<b>Arg1</b>	<Clear>  0 = Clear Working buffer 3 = Clear the L3 outgoing buffer 4 = Clear the L4 outgoing buffer 5 = Clear the L5 outgoing buffer
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>120</b>
<b>Name</b>	Test B Address in L5 incoming message equal to associated channel
<b>Description</b>	Tests if the B address in the L5 incoming message is equal to the associated channel maintained by CM. If there is no active association, a FALSE is returned.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>121</b>
<b>Name</b>	Test B Address in L5 incoming message equal to associated channel
<b>Description</b>	Tests if the B address in the L5 incoming message is equal to the associated channel maintained by CM. If there is no active association, a FALSE is returned.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Send L5 DS0 status change using L5 outgoing buffer
<b>Description</b>	Sends the specified message to L5..
<b>Arg1</b>	<Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Send L5 RFS using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<RFS Retry Flag>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Send L5 RFS w/ Data using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<RFS Retry Flag>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>133</b>
<b>Name</b>	Send L5 Channel Released using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Send L5 Channel Released w/ Data using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Send L5 Call Processing Event using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Send L5 Release Request using outgoing L5 buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>137</b>
<b>Name</b>	Send L5 Call Progress Analysis Result using L5 outgoing buffer
<b>Description</b>	Sends the specified message to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>157</b>
<b>Name</b>	Test RW channel status Interface Type
<b>Description</b>	Tests the current channel interface type. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>158</b>
<b>Name</b>	Test L3 Incoming buffer for address data flag
<b>Description</b>	Test for address data TLV in incoming L3 buffer. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>159</b>
<b>Name</b>	Store Access Denied data from L3 incoming buffer into
<b>Description</b>	Stores the access denied reason in the GPR from the L3 access denied message.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>161</b>
<b>Name</b>	Test Channel Answer Supervision Mode
<b>Description</b>	Tests the current channel answer supervision mode. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>162</b>
<b>Name</b>	Test L3 Channel state type in current message
<b>Description</b>	Tests the L3 channel status type for INS/OOS/etc.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>163</b>
<b>Name</b>	Update RW L3 data based on L3 channel status in current message and RO config bytes
<b>Description</b>	Sets up the channel's private RW properties based on the L3 provided data and the L4CH config bytes.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>164</b>
<b>Name</b>	Test L3 Incoming buffer for call active flag
<b>Description</b>	Tests the L3 incoming message for the call active flag.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>165</b>
<b>Name</b>	Test Channel Flash Timing Mode
<b>Description</b>	Tests the current channel flash timing mode. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>166</b>
<b>Name</b>	Test L3 Incoming buffer for L3 Clear Request Data
<b>Description</b>	Test for L3 clear request data TLV in incoming L3 buffer. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>170</b>
<b>Name</b>	Send L3 Outseize Control request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<GPR # containing Layer 4 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>171</b>
<b>Name</b>	Send L3 Call request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>172</b>
<b>Name</b>	Send L3 Inseize Control request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<GPR # containing Layer 4 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>173</b>
<b>Name</b>	Send L3 Flash request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>174</b>
<b>Name</b>	Send L3 User Info request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>175</b>
<b>Name</b>	Send L3 Generate L5 Event request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>176</b>
<b>Name</b>	Send L3 Transmit Off-hook request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>177</b>
<b>Name</b>	Send L3 Transmit Idle request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>180</b>
<b>Name</b>	Send L3 Alerting request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message. Arg1 is a flag which determines whether the L3 outgoing buffer should be saved il L4CH after its sent to L3. This is needed to support Connect With Data for ISDN.
<b>Arg1</b>	<Action>  0 = Remove Buffer 1 = Retain Buffer
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>181</b>
<b>Name</b>	Send L3 Connect request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>182</b>
<b>Name</b>	Send L3 Clear request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>183</b>
<b>Name</b>	Send L3 Host Connect request using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>184</b>
<b>Name</b>	Send L3 Busy Out ACK using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>185</b>
<b>Name</b>	Send L3 PPL Event using L3 outgoing buffer
<b>Description</b>	Sends the specified L3 message.
<b>Arg1</b>	<Event>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>186</b>
<b>Name</b>	Incrementing L4CH Function
<b>Description</b>	L4CH Function which increments the value in the global gp register
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Incr value>

<b>AF Number</b>	<b>187</b>
<b>Name</b>	Decrementing L4CH Function
<b>Description</b>	L4CH Function which decrements the value in the global gp register
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Decr value>

<b>AF Number</b>	<b>188</b>
<b>Name</b>	Comparing L4CH Function
<b>Description</b>	L4CH Function which compares the value in the global general purpose register
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Test value>
Note	Returns True if equal

<b>AF Number</b>	<b>189</b>
<b>Name</b>	Comparing GGPR L4CH Function
<b>Description</b>	L4CH Function which compares the value in the GGPR with GPR.
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<GPR #>
Note	Returns True if equal

<b>AF Number</b>	<b>190</b>
<b>Name</b>	Copying L4CH Function
<b>Description</b>	L4CH Function which copies the value in the GGPR to GPR.
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<GPR #>

<b>AF Number</b>	<b>191</b>
<b>Name</b>	Copying GPR L4CH Function
<b>Description</b>	L4CH Function which copies the value in the GPR to GGPR.
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Test value>

<b>AF Number</b>	<b>192</b>
<b>Name</b>	Clearing GGPR L4CH Function
<b>Description</b>	L4CH Function which clears the contents in the GGPR indicated
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>193</b>
<b>Name</b>	Clearing GGPR L4CH Function
<b>Description</b>	L4CH Function which clears a number (indicated by arg2)
<b>Arg1</b>	<GGPR #>
<b>Arg2</b>	<Range #>  194 – 200  Reserved for future Global GPR AFs

<b>AF Number</b>	<b>194-200</b>
<b>Name</b>	Reserved for future Global GPR AFs
<b>Description</b>	Reserved for future Global GPR AFs
<b>Arg1</b>	Not Applicable
<b>Arg2</b>	Not Applicable
<b>Note</b>	Reserved for future Global GPR AFs

<b>AF Number</b>	<b>201</b>
<b>Name</b>	Send Call Service Request to CM using L4 outgoing buffer
<b>Description</b>	Formats and sends a Call Service Request, including the current basic call state, to the local L4CM component .
<b>Arg1</b>	<Local Call State>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>202</b>
<b>Name</b>	Send Call Service ACK to CM using L4 outgoing buffer
<b>Description</b>	Responds to an L4CM Call Service Request with a Call Service ACK to CM including the current basic call state.
<b>Arg1</b>	<Local Call State>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>203</b>
<b>Name</b>	Send Call Service 1way Forced Request to CM using L4 outgoing buffer
<b>Description</b>	Formats and sends a Call Service Request for a 1-way forced connection to the local L4CM component, including the current basic call state.
<b>Arg1</b>	<Local Call State>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>204</b>
<b>Name</b>	Send Call Service Reject to CM with reason using L4 outgoing buffer
<b>Description</b>	Responds to an L4CM component Call Service Request with a Call Service Reject to CM. The reject reason is derived from the GPR #. If the GPR number is 0 then it is derived directly from Arg2.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Reason>

<b>AF Number</b>	<b>205</b>
<b>Name</b>	Send Alerting Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication the a local alerting was received from L3 to CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>206</b>
<b>Name</b>	Send Cut-thru Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication the a local cut-thru was received from L3 to CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>207</b>
<b>Name</b>	Send Answer Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication the a local cut-thru was received from L3 to CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>208</b>
<b>Name</b>	Send Clear Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that the local call is being released to CM. The DERM override is a flag used by CM to determine whether a clear call or clear connection is sent based on the DERM of the channel.
<b>Arg1</b>	<DERM Flag>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>209</b>
<b>Name</b>	Test CM Call State for Active Call
<b>Description</b>	Tests the CM managed shared RW for an active CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>210</b>
<b>Name</b>	Send Flash Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that a flash event was received from L3 to CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>211</b>
<b>Name</b>	Send Clear ACK Call Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that the clear event has been processed.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>212</b>
<b>Name</b>	Test call service reject reason from L4 incoming buffer
<b>Description</b>	Tests the call service reject reason in the received call service reject message from CM. Return in IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>213</b>
<b>Name</b>	Store call service reject reason from L4 incoming buffer in
<b>Description</b>	Stores the call service reject reason the specified GPR.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>214</b>
<b>Name</b>	Send Clear Connection Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that the local channel is requesting a clearing of the association with the remote channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>215</b>
<b>Name</b>	Send Clear Connection Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that the local channel is requesting a clearing of the association with the remote channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>222</b>
<b>Name</b>	Send CM Event using L4 outgoing buffer
<b>Description</b>	Sends a programmable event to CM
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>223</b>
<b>Name</b>	Send Clear Connection Service to CM using L4 outgoing buffer
<b>Description</b>	Sends an indication that the local channel is requesting a clearing of the association with the remote channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>224</b>
<b>Name</b>	Send Update Channel Status Message to CM
<b>Description</b>	Sends Update Channel Status Message to CM.
<b>Arg1</b>	<Channel Status> 1 = Free 2 = Busy
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>225</b>
<b>Name</b>	L4CH Test MCC Association
<b>Description</b>	Determine if an MCC instance is active for the local channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>226</b>
<b>Name</b>	L4CH Query Virtual L4 Type
<b>Description</b>	Determines L4 bearer type (physical TDM or VoIP, virtual TDM or VoIP) by checking localChanLogicalID
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>227</b>
<b>Name</b>	L4CH Query bearer service status
<b>Description</b>	Determines whether bearer service has been requested for this connection.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>228</b>
<b>Name</b>	L4CH Check Associated PL4
<b>Description</b>	Checks whether valid PL4 is associated with this VL4 or not.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>229</b>
<b>Name</b>	L4CH Send PL4 Request to RTR
<b>Description</b>	Obtains PL4 for use by a VL4 when a physical connection is requested.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>230</b>
<b>Name</b>	L4CH Free Up PL4
<b>Description</b>	Frees up PL4 associated with this VL4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>231</b>
<b>Name</b>	L4CH Process RTR Response
<b>Description</b>	Extracts coupled PL4 ID from router reply and stores in “coupled” location in VL4’s shared RW area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>232</b>
<b>Name</b>	L4CH Send Modify Channel Service to L3 using L3 O/G buffer
<b>Description</b>	Sends the local route an indication that the channel is available to terminate route attempts.
<b>Arg1</b>	<Channel Status> 0= Bearer Service On 1= Bearer Service Off
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>233</b>
<b>Name</b>	L4CH Check if P-L4 decoupling needed
<b>Description</b>	Checks if a virtual span/channel should be decoupled from a physical span/channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>234</b>
<b>Name</b>	L4CH Check if egress L4 is a P-L4.
<b>Description</b>	Checks if the out bound call leg is a physical span/channel or not.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>246</b>
<b>Name</b>	Send Router member available
<b>Description</b>	Sends the local route an indication that the channel is available to terminate route attempts.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>247</b>
<b>Name</b>	Send Router member busy
<b>Description</b>	Sends the local route an indication that the channel is not available to terminate route attempts because it is involved in a call.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>248</b>
<b>Name</b>	Send Router member OOS
<b>Description</b>	Sends the local route an indication that the channel is available to terminate route attempts because it is OOS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>256</b>
<b>Name</b>	L4CH Insert digit string into DSP data list from TC incoming buffer
<b>Description</b>	Copies one entire TC digit string into the DSP data list.
<b>Arg1</b>	<Data List Index>
<b>Arg2</b>	<String #>

<b>AF Number</b>	<b>257</b>
<b>Name</b>	L4CH Insert digit string into DSP data list from TC incoming buffer
<b>Description</b>	Copies one entire TC digit string into the DSP data list.
<b>Arg1</b>	<Data List Index>
<b>Arg2</b>	<GPR w/ String #>

<b>AF Number</b>	<b>258</b>
<b>Name</b>	L4CH Insert digit string into DSP data list from TC incoming buffer
<b>Description</b>	Copies one entire TC digit string into the DSP data list.
<b>Arg1</b>	<GPR w/ Data List Index>
<b>Arg2</b>	<String #>

<b>AF Number</b>	<b>259</b>
<b>Name</b>	L4CH Insert digit string into DSP data list from TC incoming buffer
<b>Description</b>	Copies one entire TC digit string into the DSP data list.
<b>Arg1</b>	<GPR w/ data list index>
<b>Arg2</b>	<GPR w/ String #>

<b>AF Number</b>	<b>260</b>
<b>Name</b>	Send <i>Collect Digit String</i> (suspend)
<b>Description</b>	Sends the appropriate <i>DSP Service Request</i> .
<b>Arg1</b>	<GPR offset>
<b>Arg2</b>	<Cfg byte offset>
Note	<p><b>Arg1</b> indicates the offset of the first GPR in the sequence to be read. This may be used for such data as:</p> <ul style="list-style-type: none"><li>mode</li><li>maximum digits</li><li># of term</li><li>config flag</li><li>termchars</li><li>address signaling type</li><li># of strings to collect</li></ul> <p>Arg2 specifies the Config Byte offset where more static data is stored. This data may include the following:</p> <ul style="list-style-type: none"><li>inter digit</li><li>first digit</li><li>completion</li><li>minimum receive duration</li><li>resume digit collection time</li><li>minimum receive inter-digit timer</li></ul> <p>This function suspends until a positive response is returned.</p>

<b>AF Number</b>	<b>261</b>
<b>Name</b>	L4CH Send <i>Collect Digit String</i> (no suspend)
<b>Description</b>	Sends the appropriate <i>DSP Service Request</i> .
<b>Arg1</b>	Arg1: <GPR offset>
<b>Arg2</b>	Arg2: <Cfg byte offset>
Note	<p><b>Arg1</b> indicates the offset of the first GPR in the sequence to be read. This may be used for such data as:</p> <ul style="list-style-type: none"> <li>mode</li> <li>maximum digits</li> <li># of term</li> <li>config flag</li> <li>termchars</li> <li>address signaling type</li> <li># of strings to collect</li> </ul> <p>Arg2 specifies the Config Byte offset where more static data is stored. This data may include the following:</p> <ul style="list-style-type: none"> <li>inter digit</li> <li>first digit</li> <li>completion</li> <li>minimum receive duration</li> <li>resume digit collection time</li> <li>minimum receive inter-digit timer</li> </ul> <p>This function suspends until a positive response is returned.</p>

<b>AF Number</b>	<b>262</b>
<b>Name</b>	Send Collect Digit String Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive.
<b>Arg1</b>	<GPR # containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>263</b>
<b>Name</b>	Retrieve L5 Reference ID and response status from message in L4 incoming buffer
<b>Description</b>	Retrieves the appropriate information from the SYM DSP service request response.
<b>Arg1</b>	<GPR # containing Layer 5 Reference ID>
<b>Arg2</b>	<GPR # containing Status>

<b>AF Number</b>	<b>264</b>
<b>Name</b>	Store DSP Service Request Type specified in L5 Incoming Buffer in a GPR
<b>Description</b>	Store the service request type of the L5 DSP Service Request message in a GPR. The GPR# is specified in arg1
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>267</b>
<b>Name</b>	Send Attach DTMF Receiver Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR which contains the configuration bits.. Arg2 specifies the config byte offset from where the following data is stored digit wait, min. rcv duration timer. This function suspends until a positive response is returned from SYM.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>268</b>
<b>Name</b>	Send Attach DTMF Receiver Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR which contains the configuration bits.. Arg2 specifies the config byte offset from where the following data is stored digit wait, min. rcv duration timer. This function suspends until a positive response is returned from SYM.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>269</b>
<b>Name</b>	Send Attach DTMF Receiver Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>270</b>
<b>Name</b>	Send Energy Detection Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the sensitivity level and reporting mode. Arg2 specifies the config byte offset from where the following data is stored: Scan duration timer, completion timer. This function suspends until a positive response is returned from SYM.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>271</b>
<b>Name</b>	Send Energy Detection Request to SYM (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the sensitivity level and reporting mode. Arg2 specifies the config byte offset from where the following data is stored: Scan duration timer, completion timer. This function does not suspend primitive
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>272</b>
<b>Name</b>	Send Energy Detection Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>273</b>
<b>Name</b>	Send Attach CPA Receiver Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR which contains the CPA class. This function suspends until a positive response is returned from SYM
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>274</b>
<b>Name</b>	Send Attach CPA Receiver Request to SYM (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR which contains the CPA class. This function does not suspend primitive.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>275</b>
<b>Name</b>	L4CH Send Attach CPA receiver request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive.
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>280</b>
<b>Name</b>	Store CPC Detection action specified in L5 incoming message in GPR
<b>Description</b>	Store the CPC action of the L5 CPC Detection message (L5 incoming buffer) in a GPR. The GPR# is specified in arg1.
<b>Arg1</b>	<GPR>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>281</b>
<b>Name</b>	Send CPC Detect Enable Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. This function suspends until a positive response is returned from SYM
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>282</b>
<b>Name</b>	Send CPC Detect Enable Request to SYM (no suspend)
<b>Description</b>	Sends a CPC Detect Enable Request to SYM (no suspend)
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>283</b>
<b>Name</b>	Send CPC Detect Enable Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. This function does not suspend primitive
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>284</b>
<b>Name</b>	Send Cancel Digit Receiver to SYM
<b>Description</b>	Sends a request to cancel any active DSP services of the specified type.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>285</b>
<b>Name</b>	Send Cancel Energy Receiver to SYM
<b>Description</b>	Sends a request to cancel any active DSP services of the specified type.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>286</b>
<b>Name</b>	Send Cancel CPA Receiver to SYM
<b>Description</b>	Sends a request to cancel any active DSP services of the specified type.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>287</b>
<b>Name</b>	Store Cancel DSP Service type specified in L5 incoming buffer in GPR
<b>Description</b>	Store the service cancel type of the L5 DSP Service Cancel message in a GPR. The GPR# is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>290</b>
<b>Name</b>	Send Connect Tone Pattern Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the pattern Id, # cycles to transmit, and event flags. This function suspends until a positive response is returned from SYM
<b>Arg1</b>	<GPR Offset (Pattern ID, # of Cycles, Event Flag)>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>291</b>
<b>Name</b>	Send Connect Tone Pattern Request to SYM (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the pattern Id, # cycles to transmit, and event flags. This function does not suspend primitive.
<b>Arg1</b>	<GPR Offset (Pattern ID, # of Cycles, Event Flag)>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>292</b>
<b>Name</b>	Send Connect Tone Pattern Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive.
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>293</b>
<b>Name</b>	Send Cancel Tone/RAN xmtr request to SYM
<b>Description</b>	Sends a request to cancel any active DSP services of the specified type.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>294</b>
<b>Name</b>	Send Connect RAN Request to SYM using L5 incoming buffer (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function suspends until a positive response is returned from SYM.
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>295</b>
<b>Name</b>	Send Connect RAN Request to SYM using L5 incoming buffer (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive.
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>296</b>
<b>Name</b>	Send Outpulse Digits Request to SYM (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the signal type, string count, string format, string mode, and events flag. Arg2 specifies the config byte offset from where the following data is stored: First digit duration, digit duration, interdigit duration, and delay duration (4 UWords). The DSP Data List is used to maintain digits and digit counts. This function suspends until a positive response is returned from SYM
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>297</b>
<b>Name</b>	Send Outpulse Digits Request to SYM (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where consecutive GPRs specify the signal type, string count, string format, string mode, and events flag. Arg2 specifies the config byte offset from where the following data is stored: First digit duration, digit duration, interdigit duration, and delay duration (4 UWORDS). The DSP Data List is used to maintain digits and digit counts. This function does not suspend primitive
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config Byte Offset>

<b>AF Number</b>	<b>298</b>
<b>Name</b>	Send Outpulse Digits Request to SYM using L5 incoming buffer
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 specifies the GPR to find the L5 reference ID of the L5 message. The data in the L5 incoming buffer is used as the DSP service request data. This function does not suspend primitive.
<b>Arg1</b>	<GPR containing Layer 5 Reference ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>300</b>
<b>Name</b>	L4CH Send Cancel 2833 Digit Rcvr to CATC
<b>Description</b>	Cancels a 2833 tone receiver service
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>301</b>
<b>Name</b>	L4CH Check for CATC DSP Req
<b>Description</b>	Checks incoming L5 message to see if it contains a CATC 2833 request
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>302</b>
<b>Name</b>	L4CH Send Attach 2833 DTMF Revr Request to CATC using L5 Incoming Buffer
<b>Description</b>	Sends an attach DTMF receiver request to CATC using data in L5 incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>310</b>
<b>Name</b>	Test Tone Xmit Generate L5 Event Flag
<b>Description</b>	Tests if the currently active DSP tone generator should have a L5 event generated upon a TC outpulsing complete event. Result through IS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>311</b>
<b>Name</b>	Translate TC Incoming buffer into L5 Outgoing buffer and send L5 Call Processing event
<b>Description</b>	Formats the correct call processing event message to L5 for the incoming TC event.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>312</b>
<b>Name</b>	Store digit from First Digit event specified in TC incoming buffer in GPR
<b>Description</b>	Stores the digit from a TC first digit event in the specified buffer into a GPR. The GPR# is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>313</b>
<b>Name</b>	Store RAN ID from TC incoming buffer into
<b>Description</b>	Stores the RAN ID from a TC RAN starting or RAN complete event into the specified GPR.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>314</b>
<b>Name</b>	Store digit info. type for timing specified in TC incoming buffer in GPR
<b>Description</b>	Stores the digit info type from a TC digits, inpulse complete timeout, or inter-digit timeout event into a GPR. The GPR # is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>315</b>
<b>Name</b>	Store digit info. type for timing specified in TC incoming buffer in GPR
<b>Description</b>	Stores the digit info type from a TC digits, inpulse complete timeout, or inter-digit timeout event into a GPR. The GPR # is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>316</b>
<b>Name</b>	Store digit string digit count from TC incoming buffer
<b>Description</b>	Stores the string's digit count from a TC digits event in the specified GPR. The string number is specified in the second GPR argument. Unknown string # causes a channel purge.
<b>Arg1</b>	<GPR to store digit count>
<b>Arg2</b>	<GPR # containing string number>

<b>AF Number</b>	<b>317</b>
<b>Name</b>	Store digit string digit from TC Incoming Buffer in GPR
<b>Description</b>	Stores a digit from a TC digits event in the specified buffer into a GPR. Arg1 holds the GPR # from where consecutive GPRs specify the string# and digit#. Arg2 specifies the GPR to store the digit. Unknown string # or digit # causes a channel purge.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<GPR to store digits>

<b>AF Number</b>	<b>318</b>
<b>Name</b>	Test digit string digit from TC incoming buffer
<b>Description</b>	Tests a digit from a TC digits event in the specified buffer into a GPR. Arg1 holds the GPR with the string#. Arg2 specifies the GPR with the digit #. Unknown string # or digit # causes a channel purge
<b>Arg1</b>	<GPR containing String #>
<b>Arg2</b>	<GPR containing digit #>

<b>AF Number</b>	<b>319</b>
<b>Name</b>	Store digit from digit with timing info specified in TC Incoming Buffer in GPR <
<b>Description</b>	Stores the digit from a TC digits event with timing info in a GPR. The GPR# is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>320</b>
<b>Name</b>	Store digit interval from TC Incoming Buffer digit with timing info
<b>Description</b>	Stores the digit interval from a TC digits event with timing info. in the specified GPR.
<b>Arg1</b>	<GPR Containing Timing Information>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>321</b>
<b>Name</b>	Store Call Progress Result from TC incoming buffer in GPR
<b>Description</b>	Stores the call progress result type in the TC incoming buffer in a GPR. The GPR # is specified in Arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>322</b>
<b>Name</b>	Store energy detect flag from TC incoming buffer in GPR
<b>Description</b>	Stores the energy flag from a TC energy detected event into a GPR. The GPR # is specified in arg1.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>323</b>
<b>Name</b>	Store energy detection duration in TC incoming buffer into
<b>Description</b>	Stores the energy detect duration data from a TC energy detected event into the specified GPR.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>324</b>
<b>Name</b>	Compare digits in digit string in TC incoming buffer
<b>Description</b>	Compares the digits from a TC digits event in the TC incoming buffer with a digit string in the config bytes. Arg1 holds the GPR # from where consecutive GPRs specify the string #, digit offset, and # of digits. Arg2 specifies the config byte offset for the stored digits.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>325</b>
<b>Name</b>	Compare digits in digit string in TC incoming buffer to GPR stored digits
<b>Description</b>	Compares the digits from a TC digits event in the TC incoming buffer with a digit string stored in GPR(s). Arg1 holds the GPR # from where consecutive GPRs specify the string #, digit offset and # digits. Arg2 specifies the GPR # for the stored digits.
<b>Arg1</b>	<GPR offset (string #, digit offset, #digits)>
<b>Arg2</b>	<GPR # for start of stored digit string>

<b>AF Number</b>	<b>326</b>
<b>Name</b>	Compare BCD digits in digit string in L3 incoming buffer
<b>Description</b>	Compares the digits from a TC digits event in the TC incoming buffer with a digit string in the config bytes. Arg1 holds the GPR # from where consecutive GPRs specify the stage #, string #, digit offset, and # of digits. Arg2 specifies the config byte offset for the stored digits.
<b>Arg1</b>	<GPR Offset>
<b>Arg2</b>	<Config byte Offset>

<b>AF Number</b>	<b>327</b>
<b>Name</b>	Compare BCD digits in digit string in L3 incoming buffer to GPR stored digits
<b>Description</b>	Compares the digits from a TC digits event in the TC incoming buffer with a digit string stored in GPR(s). Arg1 holds the GPR # from where consecutive GPRs specify the stage #, string #, digit offset and # digits. Arg2 specifies the GPR # for the stored digits.
<b>Arg1</b>	<GPR offset (stage #, string #, digit offset, #digits)>
<b>Arg2</b>	<GPR # for start of stored digit string>

<b>AF Number</b>	<b>330</b>
<b>Name</b>	Send Connect RAN Request to SYM using single RAN ID in GPR (suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where the GPRs will specify the configuration options: Config flag and Event flag. Arg2 holds the GPR # which specifies the RAN ID. This is a suspend primitive function.
<b>Arg1</b>	<GPR offset (config flag, event flag)>
<b>Arg2</b>	<GPR # containing RAN ID>

<b>AF Number</b>	<b>331</b>
<b>Name</b>	Send Connect RAN Request to SYM using single RAN ID in GPR (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 holds the GPR # from where the GPRs will specify the configuration options: Config flag and Event flag. Arg2 holds the GPR # which specifies the RAN ID. This is not a suspend primitive function.
<b>Arg1</b>	<GPR offset (config flag, event flag)>
<b>Arg2</b>	<GPR # containing RAN ID>

<b>AF Number</b>	<b>332</b>
<b>Name</b>	Send Connect Tone Pattern Request to SYM without L5 data (will NOT send CPE)
<b>Description</b>	Sends the appropriate DSP service request to SYM. The host will not be informed (via Call Processing Event) when done. Arg1 will specify the Pattern ID. Arg2 will specify the number of cycles to transmit the tone pattern. This is a suspend primitive function.
<b>Arg1</b>	Arg1:<Pattern ID>
<b>Arg2</b>	Arg2:<# cycles to transmit>

<b>AF Number</b>	<b>333</b>
<b>Name</b>	Send Connect Tone Pattern Request to SYM without L5 data (will send CPE)
<b>Description</b>	Sends the appropriate DSP service request to SYM. A Call Processing Event will be sent to the host when done. Arg1 will specify the Pattern ID. Arg2 will specify the number of cycles to transmit the tone pattern. This is a suspend primitive function.
<b>Arg1</b>	<Pattern ID>
<b>Arg2</b>	<# cycles to transmit>

<b>AF Number</b>	<b>340</b>
<b>Name</b>	Create DSP Data List buffer
<b>Description</b>	Creates a buffer for DSP Data (chaining RAN IDs or digits). Initializes to all FFs. If buffer already exists, it is re-initialized. A maximum of 64 Ran Ids can be chained.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>341</b>
<b>Name</b>	Remove DSP Data List buffer
<b>Description</b>	Cleans up the DSP Data List.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>342</b>
<b>Name</b>	Load list size from DSP Data List into GPR
<b>Description</b>	Loads the current count of dsp data items in the DSP Data List into the specified GPR.
<b>Arg1</b>	<GPR #t>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>343</b>
<b>Name</b>	Insert item into DSP Data List
<b>Description</b>	Inserts the item given in arg2 into the DSP Data List at the <i>index</i> location specified in GPR. If index is -1, the item is appended to the end of the list
<b>Arg1</b>	<Index > <item>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>344</b>
<b>Name</b>	Insert item in DSP Data List
<b>Description</b>	Inserts the item given in arg2 into the DSP Data List at the <i>index</i> location specified in GPR. If index is -1, the item is appended to the end of the list
<b>Arg1</b>	<GPR # containing index #>
<b>Arg2</b>	<Item>

<b>AF Number</b>	<b>345</b>
<b>Name</b>	Insert item in DSP Data List
<b>Description</b>	Inserts the item (given in GPR) into the DSP Data List at the <i>index</i> location specified in arg1. If index is -1, the item is appended to the end of the list
<b>Arg1</b>	< Index>
<b>Arg2</b>	<GPR # Containing Item>

<b>AF Number</b>	<b>346</b>
<b>Name</b>	Insert item in DSP Data List
<b>Description</b>	Inserts the item (given in GPR) into the DSP Data List at the <i>index</i> location specified in GPR. If index is -1, the item is appended to the end of the list.
<b>Arg1</b>	<GPR # Containing Index>
<b>Arg2</b>	<GPR # Containing Item>

<b>AF Number</b>	<b>347</b>
<b>Name</b>	Set item in DSP Data List
<b>Description</b>	Assigns the item given in arg2 to the DSP Data List at the <i>index</i> location. If index is -1, the item is appended to the end of the list.
<b>Arg1</b>	<Index>
<b>Arg2</b>	<Item>

<b>AF Number</b>	<b>348</b>
<b>Name</b>	Set item in DSP Data List
<b>Description</b>	Assigns the item given in arg2 to the DSP Data List at the <i>index</i> location (given in GPR). If index is -1, the item is appended to the end of the list.
<b>Arg1</b>	<GPR # Containing Index>
<b>Arg2</b>	<Item>

<b>AF Number</b>	<b>349</b>
<b>Name</b>	Set item in DSP Data List
<b>Description</b>	Assigns the item given in GPR to the DSP Data List at the <i>index</i> location given in arg1. If index is -1, the item is appended to the end of the list.
<b>Arg1</b>	<Index>
<b>Arg2</b>	<GPR # Containing Item>

<b>AF Number</b>	<b>350</b>
<b>Name</b>	Set item in DSP Data List
<b>Description</b>	Assigns the item given in GPR to the DSP Data List at the <i>index</i> location given in arg1. If index is -1, the item is appended to the end of the list.
<b>Arg1</b>	<GPR # Containing Index>
<b>Arg2</b>	<GPR # Containing Item>

<b>AF Number</b>	<b>351</b>
<b>Name</b>	Remove item from DSP Data List
<b>Description</b>	Removes the item at the location indicated by Arg1 from the DSP Data List.
<b>Arg1</b>	<Index>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>352</b>
<b>Name</b>	Remove item from DSP Data List
<b>Description</b>	Removes the at the location indicated by Arg1 from the DSP Data List.
<b>Arg1</b>	<GPR # Containing Index>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>353</b>
<b>Name</b>	Send Connect RAN Request to SYM using DSP Data List (suspend)
<b>Description</b>	Sends the appropriate <i>DSP Service Request</i> message to SYM. Arg1 will specify the Configuration flag option. Arg2 specifies the Event flag option. This is a suspend primitive function.
<b>Arg1</b>	<Config Flag>
<b>Arg2</b>	<Event Flag>

<b>AF Number</b>	<b>354</b>
<b>Name</b>	Send Connect RAN Request to SYM using DSP Data List (no suspend)
<b>Description</b>	Sends the appropriate DSP service request to SYM. Arg1 will specify the Configuration flag option. Arg2 specifies the Event flag option. This is NOT a suspend primitive function.
<b>Arg1</b>	<Config Flag>
<b>Arg2</b>	<Event Flag>

<b>AF Number</b>	<b>355</b>
<b>Name</b>	L4CH Delete and shift DSP data list entries
<b>Description</b>	Re-arranges the DSP data list by removing the entries starting at INDEX for COUNT.
<b>Arg1</b>	<GPR w/ index>
<b>Arg2</b>	<GPR w/ count>

<b>AF Number</b>	<b>356</b>
<b>Name</b>	Store start time stamp, get current time stamp, get duration, report start time stamp
<b>Description</b>	Re-arranges the DSP data list by removing the entries starting at INDEX for COUNT.
<b>Arg1</b>	Arg1: <Action> 0: report current time stamp 1: store the start time stamp 2: report the stored start time stamp 3: report the duration
<b>Arg2</b>	<Granularity in 100 milliseconds units>

<b>AF Number</b>	<b>357</b>
<b>Name</b>	Get TLV into GPR from Working buffer
<b>Description</b>	This function takes two arguments and gets the desired TLV (corresponding to Host Tag) from working buffer and puts into GPR
<b>Arg1</b>	<GPR Index>
<b>Arg2</b>	<Host tag>

<b>AF Number</b>	<b>358</b>
<b>Name</b>	Set TLV from GPR
<b>Description</b>	This function takes two arguments and makes an TLV and adds that to the working buffer.
<b>Arg1</b>	<GPR Index>
<b>Arg2</b>	<Host tag>

<b>AF Number</b>	<b>359</b>
<b>Name</b>	
<b>Description</b>	Makes DSP Data List TLV and puts into working buffer
<b>Arg1</b>	<GPR Index w/ DSP Data List Index (start point IN the array)>
<b>Arg2</b>	<GPR Index w/ Count (number of entries wanted to read from the array)>

<b>AF Number</b>	<b>360</b>
<b>Name</b>	Get CFG TLV Tag in GPR
<b>Description</b>	Retrieves the Config Byte at which the TLV pointer in the Private RW area is pointing to and loads the value in the GPR passed as an argument to the function call
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>361</b>
<b>Name</b>	Init CFG TLV Pointer to CFG Byte
<b>Description</b>	Initiates the TLV pointer in the Private RW area to point to the Config Byte passed as the argument.
<b>Arg1</b>	<Config Byte #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>362</b>
<b>Name</b>	Set CFG TLV Pointer to next TLV
<b>Description</b>	Increments the TLV pointer in the Private RW area to point to the next TLV in the Config Bytes.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>363</b>
<b>Name</b>	Load TLV Data in Data Block
<b>Description</b>	Get TLV data for the TLV pointer in Private RW area and loads it into the TLV Data Block in the Private RW area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>364</b>
<b>Name</b>	Add Routing Method TLV to CSR For Router
<b>Description</b>	Adds the Routing Method TLV to Call Service Request that goes to the CM
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>365</b>
<b>Name</b>	Add Routing search Key TLV to CSR For Router
<b>Description</b>	Adds the Routing Search Key TLV to Call Service Request that goes to the CM
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>366</b>
<b>Name</b>	Add Address Digits TLV to CSR For Router
<b>Description</b>	Adds the Address Digit TLV to Call Service Request that goes to the CM , argument can have 1,2 or 3. 1 for CPA Digits, 2 for ANI digits, 3 for Category
<b>Arg1</b>	<Digit Type #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>367</b>
<b>Name</b>	Add Resource Group ID TLV to CSR For Router
<b>Description</b>	Adds the Resource Group ID TLV to Call Service Request that goes to CM.
<b>Arg1</b>	<Digit Type #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>368</b>
<b>Name</b>	Load Address Digits [from TLV Data Block] in Digits Array
<b>Description</b>	Loads the Address Digits [from TLV Data Block] in to the Digits Array for further operations
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>369</b>
<b>Name</b>	Add Span/Channel TLV to CSR For Router
<b>Description</b>	Adds the Logical Span/Channel TLV to Call Service Request that goes to CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>370</b>
<b>Name</b>	Load TLV Data From TLV Data Block to GPR
<b>Description</b>	Loads the TLV Data From TLV Data Block in to a Gen. Purpose Register specified by the argument
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>371</b>
<b>Name</b>	Send Call Service Request to CM using L4 outgoing buffer with Protocol ID from Config Byte
<b>Description</b>	Sends the Call Service Request to CM using L4 outgoing buffer with Protocol ID derived from Config Byte #.
<b>Arg1</b>	<Config Byte #>
<b>Arg2</b>	<Local Call State>

<b>AF Number</b>	<b>372</b>
<b>Name</b>	Add Incoming Span/Channel TLV to CSR For Router
<b>Description</b>	Adds the Incoming Span/Channel TLV to Call Service Request that goes to CM
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>374</b>
<b>Name</b>	L4CH Transfer Data from DSP Data List to Digit Array
<b>Description</b>	This atomic function transfers digits from DSP Data List to the Digits array.
<b>Arg1</b>	<GPR # {GPR has Byte Count in it}>
<b>Arg2</b>	<GPR # {Offset within the DSP Data List}>
Note	<p><b>Arg1</b> is a GPR which has a count of digits to be copied. The count should be less than or equal to the number of digits in the DSP Data List.</p> <p>Arg2 is the offset within the Dsp Data List to start the copy.</p>

<b>AF Number</b>	<b>376</b>
<b>Name</b>	L4CH Load channel information into working buffer
<b>Description</b>	Loads L4CH channel information into working buffer
<b>Arg1</b>	<Channel Info> 0: Local 1: Connected
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>377</b>
<b>Name</b>	L4CH Add TLV from Digit Array to CSR For Router
<b>Description</b>	Adds Digit Type TLV to CSR for the TAG in GPR
<b>Arg1</b>	<GPR # {GPR has Tag Type in it}>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>378</b>
<b>Name</b>	L4CH Add Span/Channel TLV to CSR For Router with Offset
<b>Description</b>	This AF adds a Span/Channel Offset from its own channel num.
<b>Arg1</b>	<GGPR # {Span Offset}>
<b>Arg2</b>	<GGPR # {Channel Offset}>
Note	<p><b>Arg1</b> is the GGPR carrying Span Offset. If the GGPR has 0 in the offset value is taken from the cfg Byte TLV Data.</p> <p>Arg2 is the GGPR carrying Channel offset. If the GGPR has 0 in the offset value is taken from the cfg Byte TLV Data.</p>

<b>AF Number</b>	<b>379</b>
<b>Name</b>	L4CH Add Call Profile stamp
<b>Description</b>	This AF where ever inserted adds the call profiling Data for that channel if Call profiling for that channel is enabled.
<b>Arg1</b>	<Check Point Number #>
<b>Arg2</b>	<Not Used>
<b>Note</b>	Arg1 has the check point number that the call profile has to be tagged with.

<b>AF Number</b>	<b>381</b>
<b>Name</b>	L4CH Check CSRin incoming_l4 Buff, whether to send Call Req or Outseize Control
<b>Description</b>	Outgoing CH looks for the TLV inserted by incoming CH by calling atomic function 382.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Note</b>	Return 0x01 if Outseize Control

<b>AF Number</b>	<b>382</b>
<b>Name</b>	L4CH Add TLV for the call Type in CSR in L4 Outgoing Buffer
<b>Description</b>	The incoming CH instance adds this TLV for the Outgoing CH to decide whether to send L3_call_req or L3_ouseize_control.
<b>Arg1</b>	<GP Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>383</b>
<b>Name</b>	L4CH Add Standard OUTSEIZE ICB TLV in L4 incoming buffer
<b>Description</b>	L4CH adds standard OUTSEIZE ICB TLV in L4 incoming buffer
<b>Arg1</b>	<Option Type #>
<b>Arg2</b>	<Not Used>
<b>Note</b>	Adds a standard formatted ICB 1: Seize 2: Use Instruction List

<b>AF Number</b>	<b>385</b>
<b>Name</b>	L4CH Send Route Control Msg Reject to RTR using L4 outgoing buffer
<b>Description</b>	Sends NACK for the message.
<b>Arg1</b>	<GPR #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>386</b>
<b>Name</b>	L4CH Send Route Control Msg Ack to RTR using L4 outgoing buffer
<b>Description</b>	Sends an ACK for the message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>387</b>
<b>Name</b>	L4CH Store RTR Handle from Incoming L4 Buffer
<b>Description</b>	Stores the Router Handle in Private R/W for future communication with the router.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>388</b>
<b>Name</b>	L4CH Check CSR in l4 incoming, whether the CSR has Outseize ICBs
<b>Description</b>	Checks for Outseize Control ICB in the message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Note</b>	{ Returns 0-Yes; 1-No }

<b>AF Number</b>	<b>389</b>
<b>Name</b>	Invoke Timer
<b>Description</b>	Initiates the timer type specified in Arg 1 for the duration specified in the GPR indicated in Arg 2.
<b>Arg1</b>	<Timer Type>
<b>Arg2</b>	<GPR Index>

## Layer 4 Call Management (0x0062)

---

**Atomic Functions** The following Atomic Functions (AF) are specific to the Layer 4 Call Management (0x0062) PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Purge Call
<b>Description</b>	Initiates a purge of the local CH and the remote channel with reason in arg1.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Test Local Channel State of Msg in CH Incoming Buffer
<b>Description</b>	Tests the local CH channel state in either a CH Call Service Request or Call Service Ack. Return value through internal state.
<b>Arg1</b>	<Reason>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Test Channel Answer Supervision Mode
<b>Description</b>	Tests the shared RW answer supervision mode. Return value through internal state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Block Invalid Events in next Normal State
<b>Description</b>	Function to tell the state machine pre-processor to block any events which would be invalid events for the next normal state. The blocking is cleared after the state is exited.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Test Local End Release Mode
<b>Description</b>	Tests the shared RW local end release mode. Return value through internal state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Test Distant End Release Mode
<b>Description</b>	Tests the shared RW distant end release mode. Return value through internal state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Reset CM remote channel database
<b>Description</b>	Function to reset the RW area when a call is being torn down.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Note</b>	The following RW variables are reset: Remote Chan ID, Remote PCM Format and Remote DB Padding.

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Store Local Channel State in L4 CH Incoming Buffer into GPR
<b>Description</b>	Extracts the L4 CH call state from the incoming message and stores it into the specified GPR.
<b>Arg1</b>	<GPR>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Stop all CM Timers
<b>Description</b>	Stops any active CM Timers
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Test Flash Timing Mode
<b>Description</b>	Tests the shared RW flash timing mode. Return value through internal state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Test DERM Override in CH Incoming Buffer
<b>Description</b>	Tests the value of the CH provided DERM override flag in the message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	ACK PPL Event Request
<b>Description</b>	Used to ack the host PPL Event Request
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send L5 PPL Event Indication <event ID> using outgoing L5 buffer
<b>Description</b>	Sends a PPL Event Indication to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Send Connect Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a connect request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send Call Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a call service request to the remote CM. All relevant channel information is loaded into the outgoing message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send Call Service Ack Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a call service ack to the remote CM. All relevant channel information is loaded into the outgoing message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send Call Service Reject Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a call service reject to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send Alerting Call Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send an alerting request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send Cut-Thru Call Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a cut-thru request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send Answer Call Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send an answer request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Send Clear Connection Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a clear connection request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Send Clear Call Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a clear call request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send CSR Reject to source of current msg
<b>Description</b>	Function to immediately send a call service reject to the remote CM with the specified reject reason.
<b>Arg1</b>	<Reject Reason>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send Channel Information Request to CM using outgoing CM buffer
<b>Description</b>	Function to send a channel information request to the remote CM. All relevant channel information is loaded into the outgoing message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send Channel Information Ack using current CM message
<b>Description</b>	Function to immediately send a call information ack to the remote channel that requested the information.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Send Flash Service Request to CM using CM outgoing buffer
<b>Description</b>	Function to send a flash request to the remote CM.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Send Call Service Reject Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a call service reject indication to the local CH in response to a call service request made by the CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Send Call Service ACK Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a call service ack indication to the local CH in response to a call service request made by the CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Send Alerting Call Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send an alerting request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Send Cut-Thru Call Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a cut-thru request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Send Answer Call Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send an answer request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Send Call Service Request Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a call service request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Send Clear Call Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a clear call request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Send Clear Connection Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a clear connection request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Send Flash Service Indication to CH using CH outgoing buffer
<b>Description</b>	Function to send a flash request indication to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	Send Clear Ack to CH using CH outgoing buffer
<b>Description</b>	Function to send a clear ack event to the local CH.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>110</b>
<b>Name</b>	
<b>Description</b>	Send L4RTR Route Call Service Request using RTR Outgoing Buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>120</b>
<b>Name</b>	Store Remote Chan ID from Call Service Req in CH Incoming Buffer
<b>Description</b>	Function to store only the remote channel ID into the local RW area from an incoming call service request or ack for acknowledgment or reject return purposes only.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>121</b>
<b>Name</b>	Store Remote Chan Data from CM Call Service Req/Ack in CM Incoming Buffer
<b>Description</b>	Function to store all relevant channel information (i.e. Remote Chan ID, Remote PCM Format, Remote DB Padding) into the local RW area from an incoming call service request or ack.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>140</b>
<b>Name</b>	Store CH data in CH incoming buffer
<b>Description</b>	Transfers any data buffer attached to the CH message to the CH incoming buffer. Any data previously in the CH incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>141</b>
<b>Name</b>	Store CM data in CM incoming buffer
<b>Description</b>	Transfers any data buffer attached to the CM message to the CM incoming buffer. Any data previously in the CM incoming buffer is lost.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>142</b>
<b>Name</b>	Transfer data from CH incoming to CM outgoing
<b>Description</b>	Transfers data from one buffer to another in the specified mode. The buffer pointer is simply moved and the source pointer is cleared.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>143</b>
<b>Name</b>	Transfer data from CM incoming to CH outgoing
<b>Description</b>	Transfers data from one buffer to another in the specified mode. The buffer pointer is simply moved and the source pointer is cleared.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>144</b>
<b>Name</b>	Clear CH outgoing buffer
<b>Description</b>	De-allocates any CH outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>145</b>
<b>Name</b>	Clear CM outgoing buffer
<b>Description</b>	De-allocates any CH outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>146</b>
<b>Name</b>	Test for CH outgoing buffer
<b>Description</b>	Deallocates any CH incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Test for CM outgoing buffer
<b>Description</b>	Deallocates any CM incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>160</b>
<b>Name</b>	Send Local Call Connect VP Request to PC
<b>Description</b>	Informs PC that the local CM is ready for a voice path connection
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>161</b>
<b>Name</b>	Send Remote Call Connect VP Request to PC
<b>Description</b>	Informs PC that the remote CM is ready for a voice path connection
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>162</b>
<b>Name</b>	Send Disconnect VP Request to PC
<b>Description</b>	Instructs PC to disconnect the voice path
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>163</b>
<b>Name</b>	Send Local Forced Connect VP Request to PC
<b>Description</b>	Instructs PC to immediately setup the voice path connection.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>170</b>
<b>Name</b>	Set/Update Routing Status for Channel
<b>Description</b>	Updates the routing Database to the status { 1= FREE, 2=BUSY }
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Note</b>	<Status 1=FREE, 2=BUSY>

<b>AF Number</b>	<b>171</b>
<b>Name</b>	Send Call Service Reject Indication to RTR using RTR Outgoing
<b>Description</b>	Sends a Reject message to the Remote RTR in response to a Call service request.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>172</b>
<b>Name</b>	Store Remote Chan Data from RTR Call Service Req/Ack in RTR Incoming Buffer
<b>Description</b>	Stores the Remote Chan Data from RTR Call Service Req/Ack in RTR Incoming Buffer in the private RW area
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>173</b>
<b>Name</b>	Transfer CH Incoming into RTR Outgoing
<b>Description</b>	Transfer's the message from CH incoming buffer to RTR outgoing buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>174</b>
<b>Name</b>	Store RTR Data Into RTR Incoming/RTR Addr. Block
<b>Description</b>	Stores the RTR Data Into RTR Incoming/RTR Addr. Block in the private RW area
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>175</b>
<b>Name</b>	Transfer Data from RTR Incoming to CH Outgoing
<b>Description</b>	Transfer's the message from RTR incoming buffer to CH outgoing buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>176</b>
<b>Name</b>	Send Call Service Request Ack to RTR using RTR outgoing buffer
<b>Description</b>	
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

## Layer 4 Physical Connection (0x0063)

---

**Atomic Functions** The following Atomic Functions (AF) are specific to the Layer 4 Physical Connection (0x0063) PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	PCM Connect local channel to remote channel
<b>Description</b>	Establishes a 1-way PCM connection with the remote channel as the source.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Disconnect local channel
<b>Description</b>	Disconnects the local channel form all other channels.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	PCM Forced connect local channel to remote channel
<b>Description</b>	Sends a forced connection message to the local switch manager.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	ACK PPL Event Request
<b>Description</b>	Sends an acknowledgment to a host <i>PPL Event Request</i> message with the status indicated in Arg1.
<b>Arg1</b>	<Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Send L4 PPL Event Indication
<b>Description</b>	Sends a <i>PPL Event Indication</i> message to the host with the event indicated by Arg1.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

## Call Control Router (0x0064)

---

**Atomic Functions** The following AFs are specific to the CSP Call Control Router (0x0064) PPL component.

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Store incoming data in incoming buffer <0:initiate,1:terminate>
<b>Description</b>	Stores the incoming data in the buffer passed as the argument.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Get Routing method type from incoming initiate buffer and load in Gen. Purpose Register
<b>Description</b>	This function extracts the Route Group ID from the incoming message.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Load Route Group ID type search key from incoming initiate buffer in Gen. Purpose Register
<b>Description</b>	This function extracts the Route Group ID from the incoming message.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Set Resource Group Member pointer to the Address Element indexed by Gen. Purpose Register
<b>Description</b>	Points the Resource Group Member pointer to the Address Element index given in the GP Register.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Load Entry Data tag in Gen. Purpose Register
<b>Description</b>	Loads the Entry Data Tag from the TLV that Entry Data TLV Pointer currently points to.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Test for end of route Table
<b>Description</b>	Tests for end of the route table for a given row num passed as an argument to the atomic function ( Null pointer means end of the route table... It means that the route table has to be continuous with non null rows ).
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Load Resource Group Member count in Gen. Purpose Register
<b>Description</b>	Loads the count of the total members in a Resource Group, in the specified Gen. Purpose Register.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	Test address type <0:initiate,1:terminate>
<b>Description</b>	Tests the address type for the initiate and terminate addresses
<b>Arg1</b>	0-initiate address;  1-terminate address  Return Value: PPLevINT_EVENT_0 - Invalid or unrecognized address PPLevINT_EVENT_1 - Timeslot address (CM) PPLevINT_EVENT_2 - Router address (RTR)
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>111</b>
<b>Name</b>	Clear Buffer <0:incoming initiate,1:outgoing initiate,2:incoming terminate,3:outgoing terminate>
<b>Description</b>	Function to clear a specified working buffer
<b>Arg1</b>	0-incoming init. 1-outgoing init. 2-incoming term 3-outgoing term
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>114</b>
<b>Name</b>	Check Available status of timeslot in terminate address
<b>Description</b>	Checks the status of the local timeslot; returns PPLevINT_EVENT_1 if free or PPLevINT_EVENT_0 if Busy.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>115</b>
<b>Name</b>	Send Call Service Request to terminate address using outgoing terminate buffer
<b>Description</b>	Sends Call Service Request to terminate address using outgoing terminate buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>116</b>
<b>Name</b>	Send Call Service Reject to initiate address using outgoing initiate buffer
<b>Description</b>	Sends Call Service Reject to initiate address using outgoing initiate buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>117</b>
<b>Name</b>	Set Route table to point to the row number in Gen. Purpose Register
<b>Description</b>	Directs Route Table Pointer to the row number specified in the GP register.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>120</b>
<b>Name</b>	Set Entry Data TLV pointer to TLV indexed by Gen. Purpose Register
<b>Description</b>	Points the TLV pointer to the TLV index given in the GP Register.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>121</b>
<b>Name</b>	Load TLV count in Gen. Purpose Register
<b>Description</b>	Loads TLV count of the Entry Data Field of the Route Table in Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>122</b>
<b>Name</b>	Initialize the Resource Group pointer to Resource Group ID in Gen. Purpose Register
<b>Description</b>	Initializes the Resource Group pointer to Resource Group ID specified in the Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>123</b>
<b>Name</b>	Load Entry Data Field TLV Value in Gen. Purpose Register
<b>Description</b>	Loads the Entry Data Field TLV Data in Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>124</b>
<b>Name</b>	Translate Address Element Block into terminate address and validate
<b>Description</b>	<p>Validate Span/Channel address in Address Element Block and store result in Timeslot Info Block.</p> <p>Return Value:</p> <p>ERROR CASES:  PPLevINT_EVENT_0 - No Address in Block or invalid address  PPLevINT_EVENT_1 - Remote Node Busy</p> <p>SUCCESS CASES:  PPLevINT_EVENT_10 - valid local span/channel  PPLevINT_EVENT_11 - valid remote span/channel  PPLevINT_EVENT_12 - valid remote node ID</p>
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>125</b>
<b>Name</b>	Initialize Route Table pointer to the table ID from the config Byte
<b>Description</b>	Initializes Route Table pointer to the table ID specified in the config Byte.
<b>Arg1</b>	<Config Byte #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>126</b>
<b>Name</b>	Load Route Group ID from route table row in Gen. Purpose Register
<b>Description</b>	Loads the Route Group ID from route table row in the specified Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>127</b>
<b>Name</b>	Load Criteria type from route table row in Gen. Purpose Register
<b>Description</b>	Loads the Criteria Type from route table row in the specified Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>128</b>
<b>Name</b>	Load Address Digits from incoming initiate buffer into the criteria data array
<b>Description</b>	Loads Address Digits from incoming initiate buffer into the criteria data array
<b>Arg1</b>	<String #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Load incoming Logical Span/Channel num from incoming initiate buffer into the criteria data array
<b>Description</b>	Loads incoming Logical Span/Channel num from incoming initiate buffer into the criteria data array
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Load time of the day into the criteria data array
<b>Description</b>	Loads time of the day into the criteria data array
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Transfer Data in buffer <0:initiate to terminate,1:terminate to initiate> <0:move,1:copy>
<b>Description</b>	Transfer Data between buffers
<b>Arg1</b>	0-initiate to terminate 1-terminate to initiate
<b>Arg2</b>	0-move,  1-copy

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Load number of digits to compare into the Gen Purpose Register
<b>Description</b>	Loads the number of digits to compare into the specified GP Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Compare number of digits loaded in the GP Register of the criteria data array to the criteria data field in the route table (with mask
<b>Description</b>	Compares the number of digits loaded in the GP Register of the criteria data array to the criteria data field in the route table.
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>137</b>
<b>Name</b>	Compare Span Channel Address in criteria data array to the criteria data field in the route table
<b>Description</b>	Compares Span Channel Address in criteria data array to the criteria data field in the route table
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>138</b>
<b>Name</b>	Compare Time of the day in criteria data array to the criteria data field in the route table
<b>Description</b>	Compares Time of the day in criteria data array to the criteria data field in the route table.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>142</b>
<b>Name</b>	Load Resource Group Address Element TLV Data in Address Element block
<b>Description</b>	Loads the Resource Group Address Element TLV Data in Address Element block
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>148</b>
<b>Name</b>	Send Call Service Request Ack to initiate address using outgoing initiate buffer
<b>Description</b>	Sends Call Service Request Ack to initiate address using outgoing initiate buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>149</b>
<b>Name</b>	Send Call Service Reject to initiate address using outgoing initiate buffer and reject reason
<b>Description</b>	Sends Call Service Reject to initiate address using outgoing initiate buffer and reject reason
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>151</b>
<b>Name</b>	Load Resource Group TLV in Resource Group Array # in Gen Purpose Register
<b>Description</b>	Loads the Resource Group TLV in Resource Group Array # in Gen Purpose Register (Used while processing multi criteria requests)
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>152</b>
<b>Name</b>	Initialize Resource Group Table pointer to the table ID from the config Byte
<b>Description</b>	Initializes the Resource Group Table pointer to the table ID from the specified config Byte
<b>Arg1</b>	<Config Byte #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>153</b>
<b>Name</b>	Find/Test a common set of Resource Group and load them into the resource group array
<b>Description</b>	Finds/Tests a common set of Resource Group and loads them into the resource group array (Used for Multi criteria)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>154</b>
<b>Name</b>	Resource Group Table pointer to the first table ID in the resource group array
<b>Description</b>	Initializes Resource Group Table pointer to the first table ID in the resource group array
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>155</b>
<b>Name</b>	Load Resource Group TLV in Resource Group Array # in Gen Purpose Register
<b>Description</b>	Loads the Resource Group TLV in Resource Group Array # in Gen Purpose Register (Used while processing multi criteria requests)
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>156</b>
<b>Name</b>	Load Entry Data Field TLV Data in Address Element Block
<b>Description</b>	Loads Entry Data Field TLV Data in Address Element Block
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>157</b>
<b>Name</b>	Load Resource Group ID from incoming initiate buffer in Gen. Purpose Register
<b>Description</b>	Loads the Resource Group ID from incoming initiate buffer in Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>158</b>
<b>Name</b>	Load Span/Channel Address Element from incoming initiate buffer in Address Element block
<b>Description</b>	Loads the Span/Channel Address Element from incoming initiate buffer in Address Element block
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>159</b>
<b>Name</b>	Load Criteria Type search key from incoming initiate buffer into Gen. Purpose Register
<b>Description</b>	Loads the Criteria Type search key from incoming initiate buffer into Gen. Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>166</b>
<b>Name</b>	Move Gen Purpose Register to Global Gen Purpose Register < Register #>
<b>Description</b>	Moves the contents of Gen Purpose Register to the specified Global Gen Purpose Register
<b>Arg1</b>	<To Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>167</b>
<b>Name</b>	Move Global Register to Gen Purpose Register < Register #>
<b>Description</b>	Moves the contents of Global Register to the Gen Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>168</b>
<b>Name</b>	Load Resource Group ID from incoming initiate buffer in Gen. Purpose Register
<b>Description</b>	Global Register (GP Register index #) to Gen Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>169</b>
<b>Name</b>	Move Gen Register (GP Register index #) to Global Gen Purpose Register
<b>Description</b>	Moves Gen Register (GP Register index #) to Global Gen Purpose Register
<b>Arg1</b>	<Register #>
<b>Arg2</b>	<Not Used>

# 6 SS7 Atomic Functions

**Purpose** This chapter documents the atomic functions associated with the SS7 PPL components that are released for customization using the PPL Composer.

## L3P CIC (0x000F)

---

**Purpose** This section documents the atomic functions associated with L3P CIC.

**Atomic Functions** The following AFs are specific to the L3P CIC Call Control PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send L4 Channel Status In Service
<b>Description</b>	Sends L4 a channel in service indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Send ISUP Circuit Reset
<b>Description</b>	Sends ISUP a CIC reset request using no parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Fault
<b>Description</b>	Generates an SS7 card fault using fault code in Arg1.
<b>Arg1</b>	<Fault Code>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	PPL Restore Channel's Assigned Protocol
<b>Description</b>	Function to switch from the OOS to the INS protocol.
<b>Arg1</b>	<Fault Code>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	PPL Send L5 <i>PPL Event Indication</i> using L4/L5 Outgoing Buffer
<b>Description</b>	Sends a <i>PPL Event Indication</i> to the host using the event ID in Arg1 and sending up any ISUP parameters stored in the L4/L5 outgoing buffer.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send L4 a Cut Through
<b>Description</b>	Sends L4 a cut through indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send L5 PPL Event
<b>Description</b>	Sends a PPL event indication to the host using the Event ID in Arg1 and sending up ICB with the data in Arg2.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Event Data>

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send ISUP OOS Maintenance Loopback Ack
<b>Description</b>	Notifies ISUP that the requested PCM loopback has been established.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Copy Default ISUP Parameters into L5 Incoming Buffer
<b>Description</b>	Loads the PPL Config Byte prestored parameters at offset in Arg1 into the L5 incoming buffer.
<b>Arg1</b>	<Config Byte Offset of Defaults>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Test for Existence of BCD Encoded Digits in Message
<b>Description</b>	Tests an <i>Outseize Control</i> message for the existence of BCD digits.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Digits not present	0
Digits present	1

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send L4 a Q.931 Connect Message
<b>Description</b>	Sends L4 a Connect Indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send L4 a Q.931 Disconnect
<b>Description</b>	Sends L4 a disconnect Indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send L4 a Q.931 Clear using L4/L5 Outgoing Buffer
<b>Description</b>	Sends L4 a Clear indication including any parameters present in the L4/L5 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send L4 OOS Maintenance Loopback
<b>Description</b>	Send L4 an OOS indication which also requests that a PCM loopback be connected to the channel.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send L4 an Out of Service Message
<b>Description</b>	Sends an Out of Service message to L3/L4 with the reason indicated by Arg1.
<b>Arg1</b>	<Reason>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>67</b>
<b>Name</b>	Test L5 Incoming Buffer for parameters and assemble with defaults
<b>Description</b>	Search the L4/L5 incoming buffer for an ICB relating to the message configuration specified in Arg1. If found, the ICB is concatenated with prestored parameters in the Config Bytes at the index specified by Arg2 and the resulting ICB is placed into the L4 /L5 incoming buffer.
<b>Arg1</b>	<ISUP Message Config Index>
<b>Arg2</b>	<Config Byte with Default>

<b>Test Result</b>	<b>PPL Internal Event</b>
No ICB Found	0
ICB Found	1

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Copy Default ISUP Parameters and use BCD-Encoded digits into L5 Incoming Buffer
<b>Description</b>	<p>Searches for a Stage N Address Data ICB w/ BCD encoded digit strings in the L4/L5 incoming buffer. The digits are translated into SS7 Called and Calling Party Number parameters using the parameter IDs specified in the MSB (Called Party) and LSB (Calling Party) of Arg2.</p> <p>The parameters are concatenated with prestored parameters in the config bytes at the index specified in Arg1. The resulting ICB is stored in the L4/L5 incoming buffer.</p>
<b>Arg1</b>	<Config Byte Offset Of Defaults> 1-200
<b>Arg2</b>	<Called Number Party ID/Calling Number Party ID>

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Send L4 a Q.931 SETUP using L4/L5 Outgoing Buffer
<b>Description</b>	Sends L4 a call SETUP indication including any SS7 parameters present in the L4/L5 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Transfer L5 Formatted Raw SS7 Parameters from L4/L5 Incoming to ISUP Outgoing Buffer
<b>Description</b>	Searches the L4/L5 incoming buffer for the ICB corresponding to the message configuration specified by Arg1. Once found, the ICB is transferred to the ISUP outgoing buffer.
<b>Arg1</b>	<ISUP Message Config Index>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send L4 a Q.931 Alerting
<b>Description</b>	Sends L4 an alerting indication including any parameters present in the L4/L5 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Test for Existence of SS7 Parameters in Message
<b>Description</b>	Test any message for the presence of SS7 ICB buffers.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No parameters present	0
Parameters present	1

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Purge Channel
<b>Description</b>	Initiates a channel purge with purge reason specified in Arg1.
<b>Arg1</b>	<Purge Code>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send L5 PPL Event Request ACK
<b>Description</b>	Sends an acknowledgment to a PPL event request with the status specified in Arg1.

<b>Arg1</b>	<Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Assemble L5 Parameters into L5 Incoming Buffer w/ Defaults
<b>Description</b>	<p>Stores the ICBs attached to the L4/L5 message into the L4/L5 incoming buffer. A search for an ICB whose message type matches the message type of the prestored parameters specified in Arg1 is invoked.</p> <p>If found, the ICB is concatenated with prestored parameters in the Config Bytes at the index specified by Arg1 and the resulting ICB is placed into the L4 /L5 incoming buffer. If not found, the defaults are copied 'as is' into the L4 /L5 incoming buffer.</p>
<b>Arg1</b>	<Config Byte # of Defaults>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Assemble L5 Parameters into L5 Incoming Buffer w/ Defaults
<b>Description</b>	<p>Stores the ICBs attached to the L4/L5 message into the L4/L5 incoming buffer. A search for an ICB whose message type matches the message type of the prestored parameters specified in Arg1 is invoked.</p> <p>If found, the ICB is concatenated with prestored parameters in the Config Bytes at the index specified by Arg1 and the resulting ICB is placed into the L4 /L5 incoming buffer. If not found, the defaults are copied 'as is' into the L4 /L5 incoming buffer.</p>
<b>Arg1</b>	<Config Byte # of Defaults>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Transfer Network ISUP Raw SS7 Parameters from ISUP Incoming to L4/L5 Outgoing Buffer
<b>Description</b>	Transfers the ISUP incoming ICB to the L4/L5 outgoing buffer.

<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Transfer Network Raw Parameters from ISUP Incoming into BCD-Encoded Digits
<b>Description</b>	Searches the ISUP incoming buffer for the Called Party parameter (indicated by Arg1) and the Calling Party parameter (indicated by Arg2). If found, the parameters are translated into BCD encoded digits strings and placed into the L4/L5 outgoing buffer.
<b>Arg1</b>	<Called Number Party ID>
<b>Arg2</b>	<Calling Number>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send L4 a Q.931 SETUP using BCD Digits.
<b>Description</b>	Sends L4 a Call Setup using any BCD digit string in the L4/L5 outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send ISUP Setup Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP a Call Setup Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send ISUP Answer Request using ISUP Outgoing Buffer

<b>Description</b>	Sends ISUP an Answer Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Send ISUP Release Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP a Release Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Send ISUP Release Response
<b>Description</b>	Sends ISUP a Release Response using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Send ISUP Progress Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP a Progress Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Send ISUP Alert Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP an Alert Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Send ISUP Reset Response
<b>Description</b>	Sends ISUP a Reset response using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Send ISUP Maintenance Block Response
<b>Description</b>	Sends ISUP a Maintenance Block response using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Send ISUP Maintenance Unblock Response
<b>Description</b>	Sends ISUP a Maintenance Unblock response using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Send ISUP Block Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP a Block Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Send L4 OOS Span Alarm
<b>Description</b>	Sends Layer 4 an Out of Service due to a span alarm.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Send ISUP Unblock Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP an Unblock Request using the SS7 parameters present in the ISUP outgoing buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Send L4 OOS DPC Inaccessible
<b>Description</b>	Sends L4 a a channel status of OOS due to a DPC Inaccessible indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Transfer ISUP Raw SS7 Parameter from ISUP Incoming To L4/L5 Outgoing Buffer
<b>Description</b>	Copies a parameter from the incoming ISUP buffer to the outgoing L4/L5 buffer.
<b>Arg1</b>	<Parameter ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Send L5 <i>Outseize Control</i> ACK
<b>Description</b>	Acknowledges an <i>Outseize Control</i> message with status specified in Arg1.
<b>Arg1</b>	<Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Send L4 Access Denied
<b>Description</b>	Send L4 an Access Denied message with the reason specified in Arg1. The L3 Clear Pending Flag indicates if a Clear is sent to L4.
<b>Arg1</b>	<Reason>
<b>Arg2</b>	<L3 Clear Pending Flag>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Store Network SS7 Parameters into ISUP Incoming Buffer
<b>Description</b>	Stores any incoming ISUP parameters into the ISUP incoming buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Pass Return Value
<b>Description</b>	Used to request services as part of a group state machine. Arg1 is used to request different services.
<b>Arg1</b>	<Value> 1 Group Reset  2 Group Block  3 Group Unblock  4 Group Reset Response  5 Group Hardware Block  6 Group Hardware Unblock
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Store ISUP Protocol Violation ICB into L4/L5 Outgoing Buffer
<b>Description</b>	Used to store the information generated by an ISUP protocol violation in the L4/L5 outgoing buffer for transport to the host as part of a protocol violation <i>PPL Event Indication</i> .
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Store ISUP Protocol Violation ICB into L4/L5 Outgoing Buffer
<b>Description</b>	Used to store the information generated by an ISUP protocol violation in the L4/L5 outgoing buffer for transport to the host as part of a protocol violation <i>PPL Event Indication</i> .
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Send ISUP Delete CIC Event
<b>Description</b>	Sends ISUP a delete CIC request which forces a CIC to the unequipped (OOS) state. (Minimum Software Version: 5.0)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Send ISUP Connect Request using ISUP Outgoing Buffer
<b>Description</b>	Sends ISUP a connect request using the SS7 parameters present in the ISUP outgoing buffer. (Minimum Software Version: 5.0)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Take CIC Out of Service
<b>Description</b>	Takes the CIC out of service. (Minimum Software Version: 5.0)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Set Out of Service Operation Flag
<b>Description</b>	The Out of Service Operation Flag determines how the CIC will respond to the next transition from Out of Service to In service. (Minimum Software Version: 5.0)
<b>Arg1</b>	<Flag Value> 0x00 - Unblock 0x01 - Reset
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>127</b>
<b>Name</b>	Test message ID in outseize
<b>Description</b>	This atomic function returns the message ID in the <i>Outseize Control</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	PPLevINT_EVENT_1, PPLevINT_EVENT_234

<b>AF Number</b>	<b>128</b>
<b>Name</b>	Send CRM to ISUP
<b>Description</b>	Sends a CRM request to ISUP with default parameters from configuration or the parameters sent by the host
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Test ISUP CRM status.
<b>Description</b>	Test the status value in the ISUP_CRM_STATUS message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	Status event ranging from PPLevINT_EVENT_1 to PPLevINT_EVENT_255

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Send IAM request to ISUP.
<b>Description</b>	Send IAM request to ISUP after Circuit Reservation.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Cic_af_129
<b>Description</b>	Send busy out request to EX/CPU.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Atomic Function	Description	Arguments	
		Arg1 (range)	Arg2 (range)
109	Send msg. to ISUP CCR	--	--
110	Send msg. to SYM	--	--
111	Send msg. to ISUP CCO	--	--
112	Check for Positive status	--	--
113	Send msg. to ISUP DCO	--	--
114	Send ISUP CRCS ACK. msg. (ITU only)	--	--
115	Send ISUP CRCS CPA result (ITU only)	--	--
116	Send ISUP CRCS stop (ITU)	--	--
117	Send ISUP CRO stop (ANSI)	--	--
118	Send SYM Connect Tone msg.	--	--
119	Send ISUP DCO CPA result	--	--
120	Send ISUP CCO CPA result	--	--
121	Send ISUP msg. Req. using outgoing buffer	Event ID (1 : 78)	--
122	Validate and assemble msg. using config. table in L5 incoming buffer	Cfg. table entry (1 : 50)	--
123	Test message ID in PAM	--	--

124	Send L5 Event Ind. without data	Event ID. (1 : 255)	--
125	Send ISUP CVT request	msg. offset (1 : 255)	--
126	Send ISUP CVT req. for User part test	msg. offset (1 : 255)	--

<b>AF Number</b>	<b>160</b>
<b>Type</b>	Normal
<b>Name</b>	Copy CDPN digits to Digit_Buffer.
<b>Description</b>	This function will parse through the incoming setup message for the CDPN parameter. When it finds the parameter, it will copy the address signals in the parameter to the Digit_Buffer in the RW area and update the Number_of_Digits parameter in the RW area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	<Not Used>

<b>AF Number</b>	<b>162</b>
<b>Type</b>	Normal
<b>Name</b>	Append digits received in SAM to Digit_Buffer.
<b>Description</b>	This function will parse through the incoming SAM for the Subsequent Address Parameter. This will append the digits in the Subsequent Address Parameter to the Digit_Buffer in the RW area, and update the Number_of_Digits parameter to reflect the new total.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	<Not Used>

<b>AF Number</b>	<b>163</b>
<b>Type</b>	Normal
<b>Name</b>	Copy digits from Digit_Buffer to ISUP incoming buffer.
<b>Description</b>	This function will copy the digits from the Digit_Buffer to the CDPN parameter in the stored ISUP SETUP message. It will also update the CDPN parameter to reflect the increased length and also update the “Odd/Even” indicator field.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	<Not Used>

<b>AF Number</b>	<b>164</b>
<b>Type</b>	Normal
<b>Name</b>	Check the digits in the Digit_Buffer.
<b>Description</b>	This function will compare the number of digits received with the maximum number of digits to be received. It will also check whether the maximum number of digits can be collected. This is indicated by the fact that the size of the config byte is greater than the normal size of the Digit_Buffer (40 bytes).
<b>Arg1</b>	Configuration byte offset
	Input range: 0-600
<b>Arg2</b>	<Not Used>
<b>Test AF Results</b>	<Not Used>

## L3P Link (0x0010)

---

**Purpose** This section documents the atomic functions associated with L3P Link Control.

### Atomic Functions

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send Activate Link Request to MTP3
<b>Description</b>	Requests that a signaling link be activated. This attempts to bring the signaling link up, through the initial alignment procedure, to an In Service state at MTP2. This does not necessarily mean that the link is available for user traffic at MTP3 because the signaling link's inhibited status is independent of its active status.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send Deactivate Link Request to MTP3
<b>Description</b>	Requests that a signaling link be deactivated. This brings the signaling link to an inactive state at MTP2. If the signaling link was previously available at MTP3, it becomes unavailable.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Send Inhibit Link Request to MTP3
<b>Description</b>	Requests to inhibit a signaling link. An inhibited signaling link is not usable at MTP3 for user traffic. Inhibiting a signaling link does not affect its MTP2 state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send Uninhibit Link Request to MTP3
<b>Description</b>	Request to uninhibit a signaling link.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send L5 <i>PPL Event Indication</i> Message with Data
<b>Description</b>	Sends a <i>PPL Event Request</i> message to the host with data.
<b>Arg1</b>	<p>&lt;Event ID&gt;</p> <p>1 - Remote Inhibit, the signaling link has been inhibited from the remote signaling point.</p> <p>2 - Remote Uninhibit, the signaling link has been uninhibited from the remote signaling point.</p> <p>3 - Local Inhibit Denied, a local request to inhibit a signaling link has been denied, possibly due to local or remote conditions.</p> <p>4 - Local Inhibit Timeout, a local request to inhibit a signaling link has timed-out.</p> <p>5- Local Inhibit, the local request to inhibit a signaling link was successful.</p> <p>6 - Local Uninhibit, a local condition has caused a signaling link to become uninhibited.</p> <p>7 - Uninhibiting Not Possible, a request to the remote signaling point to uninhibit a signaling link was unsuccessful.</p> <p>8 - Uninhibit Timeout, a request to the remote signaling point to uninhibit a signaling link has timed-out.</p> <p>10 - Link Activated, the signaling link has become active (In Service at MTP2).</p> <p>11 - Link Dead, the signaling link is not framed properly at the physical layer (Layer 1).</p> <p>12 - L5 OOS, this acknowledges a <i>Service State Configure</i> message from the host that results in the signaling link going from the L3P active state to the L3P “L5 OOS, L1 Alive” state. A <i>Service State Configure</i> message to bring L5 to the OOS state also causes the link to be deactivated.</p>

<b>Arg2</b>	<p>&lt;Event Data&gt;</p> <p>The event data is always the normal state of the PPL component:</p> <p>0 - L5 OOS and L1 Dead</p> <p>1- L5 OOS and L1 Alive</p> <p>2 - L5 In Service and L1 Dead</p> <p>3 - L5 Alive and L1 Alive</p>
-------------	--

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send L5 a Message Response
<b>Description</b>	Generates a <i>PPL Event Request</i> Positive ACK to the host.
<b>Arg1</b>	<Status Value> 0x10 - Positive ACK
<b>Arg2</b>	<Not Used>

## ISUP CPC (0x0012)

---

**Purpose** This section documents the atomic functions associated with ISUP CPC.

### Atomic Functions

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send SPRC Circuit Status Query
<b>Description</b>	Sends a CIC circuit query to SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Test Circuit Query Local Blocking MPC Status
<b>Description</b>	Test the CIC Circuit status from SPRC for local maintenance blocking.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Circuit not locally blocked	0
Circuit locally blocked	1

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Test Circuit Query Remote Blocking MPC Status
<b>Description</b>	Test the CIC Circuit status from SPRC for remote maintenance blocking.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Circuit not remotely blocked	0
Circuit remotely blocked	1

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send SPRC CPC Circuit Status
<b>Description</b>	Sends a CIC circuit state update to SPRC with the 'CPC' status bits set to Arg1.
<b>Arg1</b>	<Circuit Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Test Parameter with Bit Mask
<b>Description</b>	<p>Tests a bit field within the parameter indicated by Arg1.</p> <p>The most significant byte of Arg2 is used as a byte offset into the parameter. The least significant byte of Arg2 is used as a bit mask which is masked with the byte in the parameter specified in the MSB of Arg2. The result is then right-shifted until the right-most bit in the mask is in the least significant bit.</p>
<b>Arg1</b>	<Parameter ID>
<b>Arg2</b>	<Byte Offset/Bit Mask>

Test Result	PPL Internal Event
Value	0 + value of bit field

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send CCI Start
<b>Description</b>	Sends a Start event to CCI.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Test Parameter for Value
<b>Description</b>	<p>Tests a byte within the parameter indicated by Arg1.</p> <p>The most significant byte of Arg2 is used as a byte offset into the parameter. The least significant byte of Arg2 is used as the test value for the byte.</p>
<b>Arg1</b>	<Parameter ID>
<b>Arg2</b>	<Byte Offset/Value>

<b>Test Result</b>	<b>PPL Internal Event</b>
Values are not equal	0
Values are equal	1

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send BLS Blocking
<b>Description</b>	Sends a Blocking event to BLS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Send SPRC MPC Remote Blk Circuit Status
<b>Description</b>	Sends a CIC circuit state update to SPRC with the 'remote blocking' status bit set to Arg1.
<b>Arg1</b>	<Circuit Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Send SPRC Release Complete
<b>Description</b>	Formats and sends a RLC message to SPRC to be submitted to MTP using L3p provided parameters (if any are present).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send CRS Start
<b>Description</b>	Sends a Start event to CRS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send SPRC Transient Circuit Status
<b>Description</b>	Sends a CIC circuit state update to SPRC with the 'transient' status bit set to Arg1.
<b>Arg1</b>	<Status>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send L3P Remote Block/Unblock Circuit Indication
<b>Description</b>	Sends L3P a Maintenance Block or Unblock, as indicated by Arg1.
<b>Arg1</b>	<Circuit Status>  0 Maintenance Block  1 Maintenance Unblock
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send L3P User-Defined Message Indication
<b>Description</b>	Formats and sends L3p a User-defined message indication including parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send SPRC User-Defined Message
<b>Description</b>	Formats and sends a user-defined message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send BLR Unblocking
<b>Description</b>	Sends an unblocking indication to BLR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>67</b>
<b>Name</b>	Send HRB Unblocking
<b>Description</b>	Sends an unblocking indication to HRB.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Send CCI Stop
<b>Description</b>	Sends a stop indication to CCI.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Resend SPRC Release
<b>Description</b>	Resends a stored REL message to SPRC to be submitted to MTP. A REL message must first be stored in the re-transmission buffer (see Atomic Functions 72 and 73).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Store IAM Information
<b>Description</b>	Temporarily stores an incoming IAM in a storage buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send L3P Setup Indication
<b>Description</b>	Formats and sends a setup indication to L3P using the parameters in the current message. If no parameters are present in the current message, then the storage buffer is used as the parameter source.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send SPRC Release
<b>Description</b>	Formats and sends a REL message to SPRC to be submitted to MTP using L3P provided parameters. The REL message is also copied and stored in the re-transmission buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send SPRC Release
<b>Description</b>	Formats and sends a REL message to SPRC to be submitted to MTP using prestored parameters in the PPL Config Bytes at offset specified in Arg1. The REL message is also copied and stored in the re-transmission buffer.
<b>Arg1</b>	<Config Byte Offset of Parameters>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send SPRC ACM
<b>Description</b>	Formats and sends an ACM message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send L3P Release Indication
<b>Description</b>	Formats and sends a release indication to L3p using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send L3P Reset Indication
<b>Description</b>	Formats and sends a reset indication to L3p using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Send SPRC ANM
<b>Description</b>	Formats and sends an ANM message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Send L3P Release Confirm
<b>Description</b>	Formats and sends a Release Confirm to L3P using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send SPRC CPG
<b>Description</b>	Formats and sends an CPG message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send SPRC IAM
<b>Description</b>	Formats and sends an IAM message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send L3P ACM Indication
<b>Description</b>	Formats and sends an ACM indication to L3P using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Send L3P CPG Indication
<b>Description</b>	Formats and sends a CPG indication to L3P using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Send L3P ANM Indication
<b>Description</b>	Formats and sends an answer indication to L3P using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Send HGBS Blocking
<b>Description</b>	Sends a blocking event to HGBS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Send L3P ISUP Event Indication
<b>Description</b>	Sends a generic event indication to L3p using the event ID specified in Arg1.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Send L5 PPL Event Indication
<b>Description</b>	Formats and sends a <i>PPL Event Indication</i> to the host with no parameters.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Test Circuit Query Local Blocking HWB Status
<b>Description</b>	Test the CIC Circuit status from SPRC for local hardware blocking.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Not Blocked	0
Blocked	1

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Validate L3P SS7 Parameters
<b>Description</b>	Function to test that all mandatory parameters and all parameters lengths are valid for a L3p provided message ICB.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Not Valid	0
Valid	1

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Test Circuit Query Remote Blocking HWB Status
<b>Description</b>	Test the CIC Circuit status from SPRC for remote hardware blocking.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Not Blocked	0
Blocked	1

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Send CRR Reset Complete
<b>Description</b>	Sends a reset complete event to CRR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Send CGRR Reset Complete
<b>Description</b>	Sends a reset complete event to CGRR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Send L3P Local Reset Indication
<b>Description</b>	Formats and sends an local reset indication to L3P.
<b>Arg1</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Send L3P Call Failure Indication with RLC pending
<b>Description</b>	Formats and sends a call failure indication to L3P with notification that a release complete event is still pending.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Send SPRC CON
<b>Description</b>	Formats and sends an CON message to SPRC to be submitted to MTP using L3P provided parameters.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Send CRS RESET After T5
<b>Description</b>	Send a 'reset because of T5' event to CRS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Send L3P CON Indication
<b>Description</b>	Formats and sends a CON indication to L3p using the parameters in the current message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Send CRCR Start
<b>Description</b>	Send a start event to CRCR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Send L3p Call Failure Indication with no RLC pending
<b>Description</b>	Formats and sends a call failure indication to L3P with notification that no release complete event is pending.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Determine Controlling Exchange Based on DPC/CIC
<b>Description</b>	Function to determine the controlling exchange for dual seizure (glare) based on ITU-TS Q.767 sec. D.2.10.1.4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	CPC Send CRI Start
<b>Description</b>	Sends a Start event to CRI.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	CPC Send L3P OOS Request
<b>Description</b>	Sends a CIC Out of Service Request to L3P.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	CPC Send L3P In Service Request
<b>Description</b>	Sends a CIC In Service Request to L3P.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Cpc_af_105
<b>Description</b>	Go through the list of optional parameters and return various internal PPL events based on PCP instruction indicator (or no PCP found) if unexpected parameters are found.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Cpc_af_106
<b>Description</b>	Discard any unknown optional parameters from the message before sending the message to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Cpc_af_107
<b>Description</b>	Send confusion if instruction indicator requires that notification should be sent.
<b>Arg1</b>	CFN Config Byte offset 150
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Cpc_af_108
<b>Description</b>	Test if notification should be sent or not.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Do not send notification	0
Send notification	1

<b>AF Number</b>	<b>110</b>
<b>Name</b>	Cpc_af_110
<b>Description</b>	Send a release with cause value 99 to the network.
<b>Arg1</b>	REL Config Byte offset 140
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>128</b>
<b>Name</b>	Send L3P CRM indication.
<b>Description</b>	Send L3P an incoming CRM indication with data in ICB format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Send L3P CRM status.
<b>Description</b>	Send the Circuit Reservation status message to L3P.
<b>Arg1</b>	Status to be sent
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Prepare ISUP message and send it to SPRC.
<b>Description</b>	Build/Re-use existing ISUP message and send it to SPRC.
<b>Arg1</b>	ISUP Message Config Index. Range 0-89
<b>Arg2</b>	Build/re-use flag Range 0-1

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Prepare ISUP message from Config Bytes and send to SPRC.
<b>Description</b>	Prepare ISUP message specified in Config Bytes and send it to SPRC.
<b>Arg1</b>	ISUP Msg. Config Index. Range 0-89
<b>Arg2</b>	PPL Config Byte offset Range 0-600

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Cpc_af_132
<b>Description</b>	Send OLM indication to L3P.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Send SSC ISUP message.
<b>Type</b>	Normal
<b>Description</b>	Send SSC an ISUP message with appropriate event.
<b>Arg1</b>	Msg Cfg Index
	0 : 89
<b>Arg2</b>	Event ID
	0 : 255
<b>Result</b>	--

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Send SSC rel/reset indication
<b>Type</b>	Normal
<b>Description</b>	Sends the one of the following events to SSC based on the event ID:  Local Reset indication  Reset indication  Call failure RLC pending  Call failure no RLC pending
<b>Arg1</b>	Event ID
	0 : 255
<b>Arg2</b>	--
<b>Result</b>	--

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Send SSC COT status
<b>Type</b>	Normal
<b>Description</b>	After the completion of the continuity procedures, this AF sends COT status to SSC.
<b>Arg1</b>	COT Status
	0 – Failure, 1 – Success
<b>Arg2</b>	None
	--
<b>Result</b>	--

<b>AF Number</b>	<b>137</b>
<b>Name</b>	Send SPRC SGM message
<b>Type</b>	Normal
<b>Description</b>	The Segmentation message constructed in the SSC is sent to the SPRC using this AF.
<b>Arg1</b>	None
	--
<b>Arg2</b>	None
	--
<b>Result</b>	--

<b>AF Number</b>	<b>138</b>
<b>Name</b>	Send SSC a PAM message
<b>Type</b>	Normal
<b>Description</b>	Forwards the PAM message to the SSC.
<b>Arg1</b>	Event ID
	0 : 255
<b>Arg2</b>	None
	--
<b>Result</b>	--

Atomic Function	Description	Arguments	
		Arg1 (range)	Arg2 (range)
104	Check Network Suspend/Resume	--	--
111	Send L3P PPL Event Ind.	Event ID (0 : 255)	--
112	Send SPRC ISUP message	Msg. cfg. index (0 : 78)	--
113	Not Used		
114	Not Used		
115	Not Used		
116	Send CCO start	--	--
117	Send CCO stop	--	--
120	Send SPRC COT status	COT Status (0 : 1)	Par. offset (0:255)
121	Send CRO start	--	--
122	Send CCO no reports	--	--
123	Test if COT expected	Par. ID (0 : 255)	Par. offset (0:255)
124	CPC send CRCS start	--	--
125	Send SPRC COT status	COT Status (0 : 1)	Par. offset (0:200)
126	Send SPRC PAM message	--	--
127	Send L5 ISUP PAM message	Event ID (0:255)	--

<b>AF Number</b>	<b>149</b>
<b>Type</b>	Normal
<b>Name</b>	Send the ISUP message to L3P CIC.
<b>Description</b>	This atomic function will send any ISUP message to L3PCIC for further processing.
<b>Arg1</b>	This is the Event ID in the L3PCIC.
	Input range: 0-255
<b>Arg2</b>	This is the ISUP Message Configuration Index of the message to be sent to L3P CIC.
	1-100
<b>Test AF Results</b>	<Not Used>

<b>AF Number</b>	<b>160</b>
<b>Name</b>	Check ISUP variant ID
<b>Description</b>	Finds ISUP variant ID of the stack.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Result</b>	Returns a PPL event corresponding to the variant ID configured.

## ISUP BLS, HGBS MGBS

---

**Purpose** The following atomic function is used by the following components:

- ISUP BLS (0x0016)
- ISUP HGBS (0x007)
- ISUP MGBS (0x0045)

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Check ISUP variant ID
<b>Description</b>	Finds ISUP variant ID of the stack.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Result</b>	Returns a PPL event corresponding to the variant ID configured.

## ISUP SPRC (0x0013)

---

**Purpose** This section documents the atomic functions associated with ISUP SPRC.

### Atomic Functions

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send Message to MTP
<b>Description</b>	Delivers an outgoing message to MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Update CPC Circuit Status and Remove Transient Circuit Status
<b>Description</b>	Updates the 'CPC' status bits and clears the 'transient' status bit in the CIC state database.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Send Circuit Status to MPC
<b>Description</b>	Returns the current circuit state from the CIC state database to the requesting ISUP state machine.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Test ISUP Message Configuration Index
<b>Description</b>	Tests for the index into the ISUP Message Configuration Table.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Index Value	0 + Index Value

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Test for Unequipped CIC
<b>Description</b>	Test that the CIC(s) that this message refers to is equipped.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Yes (Equipped)	0
No (At least one CIC is unequipped)	1

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Test If Confusion (CFN) Message
<b>Description</b>	Test if the incoming message is a CFN.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Event Data>

Test Result	PPL Internal Event
Yes	1
No	0

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send UCIC to MTP

<b>AF Number</b>	<b>57</b>
<b>Description</b>	Send a UCIC message based on the parameters stored in the configuration bytes at offset specified in Arg1.
<b>Arg1</b>	<Config Byte Offset>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send Call Control Message to CPC
<b>Description</b>	Delivers an incoming message to CPC.
<b>Arg1</b>	<Status Value>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Send L5 <i>PPL Event Indication</i> for Span/Channel
<b>Description</b>	Sends a <i>PPL Event Indication</i> message to the host for the span/channel of the current message.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Add DPC to Pause List
<b>Description</b>	Add the DPC to the pause queue and set it to expire in the number of timer ticks in Arg1.
<b>Arg1</b>	<# of Timer Ticks to Expiration>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Check for DPC in Pause List and Remove
<b>Description</b>	Tests if a DPC in a RESUME message is in the Pause queue, and if so, removes it.
<b>Arg1</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Yes (DPC is in pause list)	1
No (DPC is not in pause list)	0

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send Message to GBUR
<b>Description</b>	Delivers an incoming message to GBUR
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send Message to GBUS
<b>Description</b>	Delivers an incoming message to GBUS
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send Message to CGRR
<b>Description</b>	Delivers an incoming message to CGRR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send Message to CGRS

<b>AF Number</b>	<b>65</b>
<b>Description</b>	Delivers an incoming message to CGRS
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send Message to CRI
<b>Description</b>	Delivers an incoming message to CRI.
<b>Arg1</b>	<Status Value>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>67</b>
<b>Name</b>	Send Message to CRR
<b>Description</b>	Delivers an incoming message to CRR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Send Message to BLR
<b>Description</b>	Delivers an incoming message to BLR
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Send Message to BLS
<b>Description</b>	Delivers an incoming message to BLS
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Test if Valid Message
<b>Description</b>	Tests to determine if an ISUP Message is valid by consulting the ISUP Message Configuration template.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Yes (Valid/Known)	1
No (Invalid/Unknown)	0

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send Confusion (CFN) Message
<b>Description</b>	Send a CFN message based on the parameters stored in the configuration bytes at offset specified in Arg1.
<b>Arg1</b>	<Config Byte Offset W/ Parameters>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Test if CRS Active
<b>Description</b>	Test if CRS for the CIC is active (non-idle).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Inactive)	0
Yes (Active)	1

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Test if CRI Active
<b>Description</b>	Tests if the CRI for the CIC is active (non-idle).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Inactive)	0
Yes (Active)	1

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send MSG to CRS
<b>Description</b>	Delivers an incoming message to CRS
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Test if DPC is in Pause List
<b>Description</b>	Tests if the DPC of this message is in the DPC pause queue.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Not in queue)	0
Yes (In queue)	1

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Validate Message Parsing
<b>Description</b>	Tests that the incoming message can be parsed according to the ISUP Message Configuration template.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Corrupt message)	0
Yes (Valid message)	1

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Add MTP Message to Pause Queue
<b>Description</b>	Store outgoing message in the Pause Queue for MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Update MPC Blocking Circuit Status
<b>Description</b>	Updates the 'Maintenance Blocking' status bits (Maintenance Local Blocked, Maintenance Remote Blocked) in the CIC state database based on receiving an update message from an ISUP state machine.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Update Transient Circuit Status
<b>Description</b>	Updates the 'transient' status bit in the CIC state database based on receiving an update message from an ISUP state machine.

<b>AF Number</b>	<b>79</b>
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	De-queue an MTP Message from Pause Message Queue and Send to MTP
<b>Description</b>	Tests for queued messages for the DPC which has received a RESUME event and, if one, sends it.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (No messages in queue)	0
Yes (Message sent)	1

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Test if Pause List is Empty
<b>Description</b>	Tests if DPC Pause queue is empty.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Not Empty)	0
Yes (Empty)	1

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Decrement DPC Timer Values in Pause List
<b>Description</b>	Update all timers associated with the DPC Pause queue.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Send L3p Pause Expiration Events for Expired DPCs
<b>Description</b>	For all DPCs which have expired in Pause, queue, send a Pause event to L3p.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	De-queue all Expired DPC Messages from Pause Message Queue
<b>Description</b>	Tests for expired DPC messages in the Pause Message queue and, if any, clears them.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
No (No expired messages)	0
Yes (Expired messages)	1

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Remove All Expired DPCs from the Pause List
<b>Description</b>	Clears all expired DPCs in the DPC Pause List.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Test MTP Status Cause Byte
<b>Description</b>	Tests the status cause byte of an MTP indication.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Congestion	1
Congestion	2
Congestion	3
User Part Unavailable	4
User Part Unequipped	5
User Part Inaccessible	6

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Send L5 Network Congestion Level Alarm
<b>Description</b>	Generates an alarm to the host indicating a level of network congestion and the DPC affected.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Send L3P ISUP Unavailable Event for DPC
<b>Description</b>	Sends L3P an indication that MTP has declared a remote ISUP unavailable.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Send L3P ISUP Resume Event for DPC
<b>Description</b>	Sends L3P an indication that MTP has issued a Resume for a DPC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Send All ISUP SMs for a CIC a Stop Event
<b>Description</b>	Broadcasts a STOP event to all ISUP state machines for a CIC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Set Unequipped Bit and clear all other status for CIC

<b>AF Number</b>	<b>91</b>
<b>Description</b>	Resets the CIC state database by clearing all status bits and setting the 'unequipped' bit.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Send Message to UCIC
<b>Description</b>	Delivers an incoming message to UCIC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Update HW Blocking Circuit Status
<b>Description</b>	Updates the 'Hardware Blocking' status bits (Hardware Local Blocked, Hardware Remote Blocked) in the CIC state database based on receiving an update message from an ISUP state machine.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Update Maintenance SM Circuit Status
<b>Description</b>	Updates the maintenance-related status bits (Maintenance Unblock Wait, Hardware Unblock Wait, CRCR Recheck Active) in the CIC state database based on receiving an update message from an ISUP state machine.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Send MSG to CRCR

<b>AF Number</b>	<b>95</b>
<b>Description</b>	Delivers an incoming MSU to CRCR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Test Circuit Status for CRCR Recheck
<b>Description</b>	Tests if CRCR Recheck is active for a CIC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
No (Inactive)	0
Yes (Active)	1

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Test Message Parameter with Bit Mask
<b>Description</b>	<p>Tests a bit field within the parameter indicated by Arg1.</p> <p>The most significant byte of Arg2 is used as a byte offset into the parameter. The least significant byte of Arg2 is used as a bit mask which is masked with the byte in the parameter specified in the MSB of Arg2. The result is then right-shifted until the right-most bit in the mask is in the least significant bit</p>
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Value	0 + value of bit field

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Send Message to MGBR
<b>Description</b>	Delivers an incoming message to MGBR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Send Message to MGBS
<b>Description</b>	Delivers an incoming message to MGBS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Send Message to HGBR
<b>Description</b>	Delivers an incoming message to HGBR.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Send Message to HGBS
<b>Description</b>	Delivers an incoming message to HGBS.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Test Circuit Status for Transient
<b>Description</b>	Tests the 'transient' status bit in the CIC state database.
<b>Arg1</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
Bit not set	0
Bit set	1

<b>AF Number</b>	<b>103</b>
<b>Name</b>	CPC Circuit Status and remove unequipped status
<b>Description</b>	Updates the 'CPC' status bits and clears the 'unequipped' status bit in the CIC state database.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Clear status bits for unequipped circuits. Test if number of status bits is range
<b>Description</b>	<p>Tests if the range field of the Range and Status parameter of an incoming message is in the range defined by Arg1 and Arg2.</p> <p>The status bits for all unequipped circuits from the status field in the Range and Status parameter are cleared.</p>
<b>Arg1</b>	<Value 1> Beginning of range.
<b>Arg2</b>	<Value 2> End of range.

<b>Test Result</b>	<b>PPL Internal Event</b>
In range	1
Not in range	0

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Send L5 <i>PPL Event Indication</i> for stack
<b>Description</b>	Sends a <i>PPL Event Indication</i> to the host with entity of stack and event given by Arg1.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Place DPC in UPU queue
<b>Description</b>	Places the DPC given in the MTP indication in the UPU queue.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Check for DPC in UPU Queue and remove
<b>Description</b>	Tests the User Part Unavailable (UPU) queue for the DPC given in the MTP indication and, if there, removes it.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
DPC in queue	1
No DPC in queue	0

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Send L3P User Part Accessible
<b>Description</b>	Sends L3P an indication that a remote ISUP has become accessible.
<b>Arg1</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	Clear Unequipped Bit for CIC
<b>Description</b>	Clears the 'unequipped' status bit in the CIC state database.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>110</b>
<b>Name</b>	Update CPC Circuit Status
<b>Description</b>	Updates the 'CPC' status bits in the CIC state database.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Sprc_af_129
<b>Description</b>	Find MCP parameter in the unrecognized message.
<b>Arg1</b>	<Event ID>
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
If parameter found	1
If parameter not found	0

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Sprc_af_130
<b>Description</b>	Send unrecognized message PPL event indication containing unrecognized message to the host, with channel AIB.

<b>AF Number</b>	<b>130</b>
<b>Arg1</b>	<Event Indication ID> 0-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Sprc_af_130
<b>Description</b>	Send a release with cause value 97 to the network and send and indication to CPC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>133</b>
<b>Type</b>	Test
<b>Name</b>	SPRC test if Host controlled continuity enabled
<b>Description</b>	Looks into the common CPC area and checks if the Host Controlled Continuity is enabled.
<b>Arg1</b>	None
<b>Arg2</b>	None
<b>Test AF Results</b>	PPLevIntEVENT_0,PPLevIntEVENT_1

## ISUP SSC (0x0085)

---

**Purpose** This section documents the atomic functions associated with ISUP SSC.

### Atomic Functions

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Send ISUP message to CPC
<b>Type</b>	Normal
<b>Description</b>	Sends message with the event specified in arg1 to CPC
<b>Arg1</b>	Event
	0 : 255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Test parameter with bitmask
<b>Type</b>	Test
<b>Description</b>	Test the parameter provided in arg1 with the bit-mask in the in arg2.
<b>Arg1</b>	Parameter ID
	0 : 255
<b>Arg2</b>	Bitmask
	0 : 255
<b>Result</b>	PPLevIntEVENT_1 – If Bit mask qualifies PPLevIntEVENT_0 – If it does not

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Store Message in buffer
<b>Type</b>	Normal
<b>Description</b>	Store the message in the MSG buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Assemble message in buffer
<b>Type</b>	Normal
<b>Description</b>	Assemble the message in MSG and SGM buffer into the MSG buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send SGM message to L3P
<b>Type</b>	Normal
<b>Description</b>	Send the incoming Segmentation Message to L3P.
<b>Arg1</b>	Event ID
	0 : 255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send ISUP message to L3P
<b>Type</b>	Normal
<b>Description</b>	Send the incoming ISUP message to L3P.
<b>Arg1</b>	Event ID
	0:255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send Pass-through message to L5
<b>Type</b>	Normal
<b>Description</b>	Forward Pass through message to L5.
<b>Arg1</b>	ISUP message config index
	0 :255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Forward Incoming Message to L3P
<b>Type</b>	Normal
<b>Description</b>	Forwards Call control messages to L3P.
<b>Arg1</b>	Event
	0 : 255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Forward Message to CPC
<b>Type</b>	Normal
<b>Description</b>	This messages forwards the message to CPC without any changes to it.
<b>Arg1</b>	Event
	0: 255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send the Segmented Message to CPC
<b>Type</b>	Normal
<b>Description</b>	The AF sends the constructed SGM message to CPC.
<b>Arg1</b>	Event ID
	0:255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Store message in SGM buffer
<b>Type</b>	Normal
<b>Description</b>	Store incoming SGM message in SGM buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Test COT check flag in CPC
<b>Type</b>	Normal
<b>Description</b>	Test COT wait before RFS flag in CPC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

## L3P TUP (0x0011)

---

**Purpose** This section documents the atomic functions associated with L3P TUP.

### Atomic Functions

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Check if segmentation required
<b>Type</b>	Test
<b>Description</b>	Based on the Length of the message and the OFCI/OBCI segmentation indicator, a decision is made if the message has to be segmented.
<b>Arg1</b>	OFCI/OBCI parameter ID
	0 : 255
<b>Arg2</b>	None
	--
<b>Result</b>	PPLevIntEVENT_0 – Msg not segmented  PPLevIntEVENT_1 – Msg segmented

**TUP Virtual CIC** The following AFs are specific to the L3P TUP Call Control PPL component for TUP Virtual CIC.

<b>AF Number</b>	<b>73</b>
<b>Type</b>	Normal
<b>Description</b>	Send PPL Event Indication to Host with VCIC AIB
<b>Arg1</b>	Event Id 1:65535
<b>Arg2</b>	Not Used

<b>AF Number</b>	<b>136</b>
<b>Type</b>	Normal
<b>Description</b>	Send channel status change PPL Event Indication to Host with VCIC AIB
<b>Arg1</b>	OOS flag. 0:1
<b>Arg2</b>	OOS reason. 0:255

<b>AF Number</b>	<b>137</b>
<b>Type</b>	Normal
<b>Description</b>	Send PPL Event Request ACK with VCIC AIB
<b>Arg1</b>	Status. 0:511
<b>Arg2</b>	Not Used

## L3P BT IUP (0x0011)

---

**Purpose** This section documents the atomic functions associated with L3P BT IUP.

<b>AF Number</b>	<b>160</b>
<b>Name</b>	Send L3 an equip circuit request.
<b>Description</b>	This function will send an equip message to L3 (i.e. iupcpc).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message is queued for L3
<b>Returns</b>	Success always

<b>AF Number</b>	<b>161</b>
<b>Name</b>	Out of Service atomic function to restore the In-Service Protocol
<b>Description</b>	This function will restore the Primitive table and Event tables to point to the In-Service protocol, and the Protocol ID to the In-Service Protocol ID. The function exits with a status allowing the calling function, in message preprocessor file, to force the In-Service protocol to the desired state.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	See Description.
<b>Returns</b>	PPLerrSET_SM_STATE

<b>AF Number</b>	<b>162</b>
<b>Name</b>	Out of Service atomic function to restore the In-Service Protocol
<b>Description</b>	This function will check for congestion in the outbound direction.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPLevINT_EVENT_1 if Output Congestion is detected; PPLevINT_EVENT_0 otherwise
<b>Returns</b>	Success always

<b>AF Number</b>	<b>170</b>
<b>Name</b>	Validate L4/L5 Message
<b>Description</b>	This function will validate any message which originates from L4 or L5. Messages which contain IUP ICBs are validated and completed using the default information contained in the configuration byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	The internal “down working” buffer will contain a valid TLV formatted message, if Success is returned.  The PPLevINT_EVENT_1 is set if successful, otherwise PPLevINT_EVENT_0 is returned.
<b>Returns</b>	Success always

**AF Number**      **171**  
**Name**              Send PPL Event Indication SSM to L5  
**Arg1**                <Not Used>  
**Arg2**                <Not Used>  
**Returns**             Success always

**AF Number**      **172**

<b>Name</b>	Send PPL Event Indication SER to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>173</b>
<b>Name</b>	Send PPL Event Indication ECM to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>174</b>
<b>Name</b>	Send PPL Event Indication CFC to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>175</b>
<b>Name</b>	Send PPL Event Indication OOR to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used><Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>176</b>
<b>Name</b>	Send PPL Event Indication HLR to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>177</b>
<b>Name</b>	Send PPL Event Indication SIM to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always
<b>AF Number</b>	<b>178</b>
<b>Name</b>	Send REL with Reason in REG to L3
<b>Arg1</b>	Register number containing the Release Reason
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always
<b>AF Number</b>	<b>179</b>
<b>Name</b>	Send CAN with Reason in REG to L3
<b>Arg1</b>	Register number containing the Release Reason
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always
<b>AF Number</b>	<b>180</b>
<b>Name</b>	Set Timer associated value
<b>Description</b>	Associates a value with one of the three timers. Later when the timer expires the value may be retrieved to determine the current usage of the timer and take the appropriate action.
<b>Arg1</b>	Timer number Range 1 thru 3
<b>Arg2</b>	Associated value
<b>Returns</b>	Success always
<b>AF Number</b>	<b>181</b>

<b>Name</b>	Get Timer associated value
<b>Description</b>	Retrieves the associated value
<b>Arg1</b>	Timer number
<b>Arg2</b>	Register to contain associated value
<b>Returns</b>	Success always

<b>AF Number</b>	<b>182</b>
<b>Name</b>	Send PPL Event Indication CLR to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>183</b>
<b>Name</b>	Send PPL Event Indication RE-ANSWER (RAN) to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>184</b>
<b>Name</b>	Send PPL Event Indication SUS to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>185</b>
<b>Name</b>	Send PPL Event Indication RES to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

**Returns** Success always

**AF Number** 186

**Name** Send PPL Event Indication SWAP to L5

**Arg1** <Not Used>

**Arg2** <Not Used>

**Returns** Success always

**AF Number** 187

**Name** Send Block Upon Release Event to L3

**Arg1** <Not Used>

**Arg2** <Not Used>

**Returns** Success always

<b>AF Number</b>	<b>200</b>
<b>Name</b>	Send IAM to L3
<b>Description</b>	This function will send the IAM message to L3. The IAM message is either taken from the “down working” buffer or if the down working buffer is empty, the IAM is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>201</b>
<b>Name</b>	Send IFAM to L3
<b>Description</b>	This function will send the IFAM message to L3. The IFAM message is either taken from the “down working” buffer or if the down working buffer is empty, the IFAM is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>

<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>202</b>
<b>Name</b>	Send FAM to L3
<b>Description</b>	This function will send the FAM message to L3. The FAM message is either taken from the “down working” buffer or if the down working buffer is empty, the FAM is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>203</b>
<b>Name</b>	Send SAM to L3
<b>Description</b>	This function will send the SAM message to L3. The SAM message is either taken from the “down working” buffer or if the down working buffer is empty, the SAM is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>204</b>
<b>Name</b>	Send REL to L3
<b>Description</b>	This function will send the REL message to L3. The REL message is either taken from the “down working” buffer or if the down working buffer is empty, the REL is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>205</b>
<b>Name</b>	Copy up working buffer to save buffer.
<b>Description</b>	This function will free the save buffer, if required. Allocate a new save buffer and copy the up working buffer into the save buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>206</b>
<b>Name</b>	Append up working buffers' TLVs onto save buffer.
<b>Description</b>	This function will check to see that a save buffer has been allocated. (If not, this function will cause a channel purge.) Copy any TLVs in the up working buffer to the save buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success or PPLerrAF_INITIATED_PURGE

<b>AF Number</b>	<b>207</b>
<b>Name</b>	Swap up working buffer ptr with save buffer ptr
<b>Description</b>	This function will just sway the up working buffer ptr with the save buffer ptr.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>208</b>
<b>Name</b>	Send SND to L3
<b>Description</b>	This function will send the SND message to L3. The SND message is either taken from the “down working” buffer or if the down working buffer is empty, the SND is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>209</b>
<b>Name</b>	Send CNA with Reason to L3
<b>Description</b>	This function will send a copy of the CNA message stored in the config byte area to L3. Arg1 is used to specify the reason value.
<b>Arg1</b>	Reason
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>210</b>
<b>Name</b>	Send ACM to L3
<b>Description</b>	This function will send the ACM message to L3. The ACM message is either taken from the “down working” buffer or if the down working buffer is empty, the ACM is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>211</b>
<b>Name</b>	Send ACI to L3. The “down working “ buffer must be contain the ACI message to be sent.
<b>Description</b>	This function will send the ACI message to L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success if the down working buffer contains a message. Otherwise, the channel will be purged.

<b>AF Number</b>	<b>212</b>
<b>Name</b>	Send REL with Reason to L3
<b>Description</b>	This function will send the REL message to L3 with Arg1 containing the reason.
<b>Arg1</b>	Reason
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>213</b>
<b>Name</b>	Send ANS to L3
<b>Description</b>	This function will send the ANS message to L3. The ANS message is either taken from the “down working” buffer or if the down working buffer is empty, the ANS is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>214</b>
<b>Name</b>	Send UNDEFINED message to L3
<b>Description</b>	This function will send the a message contained in the Generic TLV Entry Data or a message beginning with a unknown H0,H1 value within the H0,H1 Element TLV to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success if the down working buffer contains a message. Otherwise, the channel will be purged.

<b>AF Number</b>	<b>215</b>
<b>Name</b>	Send ACI to L3 of user defined type
<b>Description</b>	This function will send the specified ACI type message from the config byte area to L3.
<b>Arg1</b>	Desired ACI message type. Range 1 through 7
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success if a valid ACI message type is specified. Otherwise, the channel will be purged.

<b>AF Number</b>	<b>216</b>
<b>Name</b>	Load requested ACI info from Config Bytes.
<b>Description</b>	<p>This function will send the requested ACI type message from the config byte area to L3. The requested info is expected in an ACI type 7 message contained in the up working buffer.</p> <p>Supported ACI Information Request Codes are 1 through 9 , 11, and 12.</p>
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPLvINT_EVENT_1 if the ACI message is a type 7 Information Request. Otherwise, it will return PPLvINT_EVENT_0.

<b>Returns</b>	Success if a valid ACI message type is specified. Otherwise, the channel will be purged.
----------------	--

<b>AF Number</b>	<b>217</b>
<b>Name</b>	Copy L3 message into Up Working Buffer
<b>Description</b>	This function will copy an incoming message form L3 into the up working buffer for use by L3P and possibly higher layers.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>218</b>
<b>Name</b>	Send BLO to L3
<b>Description</b>	This function will send a Block message to L3. . The BLO message is either taken from the “down working” buffer or if the down working buffer is empty, the BLO is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>219</b>
<b>Name</b>	Send UBL to L3
<b>Description</b>	This function will send a Unblock message to L3. The UBL message is either taken from the “down working” buffer or if the down working buffer is empty, the UBL is taken from the Configuration Byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>220</b>
<b>Name</b>	Send an ACI type 7 Request message to L3 with user defined IRC type
<b>Description</b>	This function will send an ACI type 7 message to L3 with the Information Request Code specified in Arg1.
<b>Arg1</b>	The Information Request Code. Range 1 through 12 <b>Except 3, 4, 10.</b>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success if Arg 1 is valid. Otherwise, the channel is purged.

<b>AF Number</b>	<b>221</b>
<b>Name</b>	Send PPL Event Indication to the Host with event specified in the register.
<b>Description</b>	This function will send a PPL Event Indication to the Host. Arg1 specifies the register containing the event.
<b>Arg1</b>	The Register number which contains the event to be reported.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>222</b>
<b>Name</b>	Set OOS op Flag to the user specified value
<b>Description</b>	This function will set the Out Of Service operation flag.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>223</b>
<b>Name</b>	L3PIUP Take CIC OOS
<b>Description</b>	This function will Cause an L3P reset to occur.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	With a status of PPLerrAF_INITIATED_RESET

<b>AF Number</b>	<b>224</b>
<b>Name</b>	L3PIUP Test OOS OP Flag
<b>Description</b>	This function will setup a PPL internal event corresponding to the value of the Operation Flag.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	With a status of PPLevINT_EVENT_0 + the value of the Operation Flag

<b>AF Number</b>	<b>225</b>
<b>Name</b>	L3PIUP Send Unequip message to L3
<b>Description</b>	This function will send an unequip message to L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>226</b>
<b>Name</b>	L3PIUP Generated In-Service event
<b>Description</b>	This function will send a Transition to In-Service message to L3P.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>227</b>
<b>Name</b>	Send A CFN Event to L3
<b>Description</b>	This function will send a CFN event to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>229</b>
<b>Name</b>	Send A PPL Event Indication of ANS received before ACM to L5
<b>Description</b>	This function will send the above PPL Event Indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>230</b>
<b>Name</b>	Send Access Denied to L4
<b>Description</b>	This function will send the Access Denied message to L4.
<b>Arg1</b>	Reason for denying access
<b>Arg2</b>	Call Status 0=Call Inactive, 1=Call Active
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>231</b>
<b>Name</b>	Send PPL Event Indication to L5 if enabled by the corresponding Config Byte.
<b>Description</b>	This function will send a PPL Event Indication to L5 if the corresponding Config byte has been enabled to allow a PPL Event Indication to be issued.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success and PPLevINT_EVENT_1 PPL Event Indication was sent. Otherwise, it is a PPLevINT_EVENT_0.

<b>AF Number</b>	<b>232</b>
<b>Name</b>	Test if IUP ICB is Present
<b>Description</b>	This function will test to see if an IUP ICB is present in the message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success and a PPLevINT_EVENT_1. Otherwise, it is a PPLevINT_EVENT_0.

<b>AF Number</b>	<b>233</b>
<b>Name</b>	Test if Stage N Address Data ICB is present
<b>Description</b>	This function will test to see if a Stage N Address Data ICB is present in the message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success and a PPLvINT_EVENT_1. Otherwise, it is a PPLvINT_EVENT_0.

<b>AF Number</b>	<b>234</b>
<b>Name</b>	Send Cut Through to L4
<b>Description</b>	This function will send a Cut Through message to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>235</b>
<b>Name</b>	Send Connect to L4
<b>Description</b>	This function will send a Connect message to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used><Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>236</b>
<b>Name</b>	Send Disconnect to L4
<b>Description</b>	This function will send a Disconnect message to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>237</b>
<b>Name</b>	Send Clear Request to L4
<b>Description</b>	This function will send a Clear Request message to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>238</b>
<b>Name</b>	Test and Get Call Type ICB
<b>Description</b>	This function will test to see if a Call Type ICB is present in the message and if so, get the call type byte.
<b>Arg1</b>	Register number to contain the Call Type byte if one is present
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPLvINT_EVENT_0 = No Call Type byte present, PPLvINT_EVENT_1 = Call Type byte is present.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>239</b>
<b>Name</b>	Setup Multi Purpose Timer 18 based upon No. Digits Requested and the Exchange Type.
<b>Description</b>	This function will setup the timeout value for timer 18 to a value based upon the number of digits required (in the register specified by Arg1) and the Exchange Type (in the register specified by Arg2).
<b>Arg1</b>	A register containing the number of digits requested in a SND message
<b>Arg2</b>	A register containing the Exchange Type. 0 = Terminating, 1 = Intermediate
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success if valid input arguments. Otherwise, the channel is purged.

<b>AF Number</b>	<b>240</b>
<b>Name</b>	Test and Get ptr to TLV in specified buffer
<b>Description</b>	This function will examine the up or down working buffer to see if the IUP ICB contains the requested TLV.
<b>Arg1</b>	The requested TLV ID
<b>Arg2</b>	The buffer to check 0 = Upworking Buffer, 1 = Downworking Buffer
<b>Outputs</b>	PPLevINT_EVENT_1 if the TLV is located and an internal variable is setup pointing to the located TLV. Otherwise, PPLevINT_EVENT_0 is returned.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>241</b>
<b>Name</b>	Test and Get TLV byte into register
<b>Description</b>	This function will get the specified byte from the working buffer containing a TLV of interest, and copy it into the specified register.
<b>Arg1</b>	The number of the byte to fetch (Range 1 to 256), Beginning with the first data byte.
<b>Arg2</b>	The number of the register to contain the requested byte
<b>Outputs</b>	If successful PPLvINT_EVENT_1 is returned, Otherwise, the specified register is cleared and PPLvINT_EVENT_0 is returned.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>242</b>
<b>Name</b>	Get Config byte into GPR 10
<b>Description</b>	<p>This function will get the specified configuration byte into General Purpose Register 10.</p> <p>The user specifies the configuration byte by Block number and Byte number.</p>
<b>Arg1</b>	The Block number containing the byte. (Range 1 to 3)
<b>Arg2</b>	The offset to the desired byte. (Range 1 to 200)
<b>Outputs</b>	If successful PPLvINT_EVENT_1, otherwise PPLvINT_EVENT_0 is returned.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>243</b>
<b>Name</b>	$\text{Arg1:GPR} = \text{Arg1:GPR} - \text{Arg2:GPR}$
<b>Description</b>	This function will subtract the contents of the register specified in Arg2 from the contents of the register specified in Arg1. The results are stored in register specified in Arg1.
<b>Arg1</b>	Any GPR
<b>Arg2</b>	Any GPR
<b>Outputs</b>	<p>PPLvINT_EVENT_0 if the resulting contents of GPR specified in Arg1 is <math>&lt; 0</math> (i.e. <math>\text{Arg2} &gt; \text{Arg1}</math>)</p> <p>PPLvINT_EVENT_1 if the resulting contents of GPR specified in Arg1 is <math>&gt; 0</math> (i.e. <math>\text{Arg1} &gt; \text{Arg2}</math>)</p> <p>PPLvINT_EVENT_2 if the resulting contents of GPR specified in Arg1 is 0 (i.e. <math>\text{Arg1} = \text{Arg2}</math>)</p>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>244</b>
<b>Name</b>	Form SND message using GPR value for N
<b>Description</b>	This message will format a SND message in TLV format using the contents of the register specified in Arg1 as the N value. (i.e. the number of digits requested). The SND message is either taken from the “down working” buffer or if the down working buffer is empty, the SND is taken from the Configuration Byte area.
<b>Arg1</b>	The number of a register containing the number of digits requested.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>245</b>
<b>Name</b>	Append accumulated addressed digits into Up Working buffer.
<b>Description</b>	This function will insert the original IAM message into the upworking buffer with the total accumulated address signals from the original IAM and subsequent SAM or FAM message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success, If a previous IAM was not received the channel is purged.

<b>AF Number</b>	<b>246</b>
<b>Name</b>	$\text{Arg1:GPR} = \text{Arg1:GPR} + \text{Arg2:GPR}$
<b>Description</b>	This function will add the contents of the register specified in Arg2 to the contents of the register specified in Arg1. The results are stored in register specified in Arg1.
<b>Arg1</b>	Any GPR
<b>Arg2</b>	Any GPR
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>247</b>
<b>Name</b>	Save Address Digits
<b>Description</b>	This function will save the received address digits (form a IAM or IFAM) into an internal buffer. The contents of the internal buffer are overwritten with the new address digits.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>248</b>
<b>Name</b>	Append Address Digits
<b>Description</b>	This function will append the digits received from a SAM or FAM to internal buffer containing previously saved address digits.
<b>Arg1</b>	The number of a General Purpose Register to contain the total number of address digits collected for this call.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success, Otherwise, PPLerrAF_INITIATED_PURGE if no digits were previously saved.

<b>AF Number</b>	<b>249</b>
<b>Name</b>	Form ICB Setup message for L4
<b>Description</b>	This function will form a setup message for L4 in ICB format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success or PPLerrAF_INITIATED_PURGE if up working buffer is not valid.

<b>AF Number</b>	<b>250</b>
<b>Name</b>	Form BCD Setup message for L4
<b>Description</b>	This function will form a setup message for L4 in BCD format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success or PPLerrAF_INITIATED_PURGE if up working buffer is not valid.

<b>AF Number</b>	<b>251</b>
<b>Name</b>	Send Formatted Setup message to L4
<b>Description</b>	This function will send the formatted message created by (Form ICB Setup message for L4 or Form BCD Setup message for L4) and send it to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>252</b>
<b>Name</b>	Send PPL Event Indication ACI to L5
<b>Description</b>	This function will send a PPL Event Indication of ACI to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>253</b>
<b>Name</b>	Send PPL Event Indication SAM to L5
<b>Description</b>	This function will send a PPL Event Indication of SAM to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>254</b>
<b>Name</b>	Send PPL Event Indication FAM to L5
<b>Description</b>	This function will send a PPL Event Indication of FAM to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>255</b>
<b>Name</b>	Send NACK or ACK to L5
<b>Description</b>	This function will send a ACK or NACK to L5 based upon the contents of Arg1.
<b>Arg1</b>	0 = NACK, 1 = ACK
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>256</b>
<b>Name</b>	L3PIUP Send Response to Outseize Control Message
<b>Description</b>	This function will send an response to L5 with a status specified in Arg1.
<b>Arg1</b>	The value to be sent in the status field of the message
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>257</b>
<b>Name</b>	Send PPL Event Indication UNDEFINED Msg to L5
<b>Description</b>	This function will send a PPL Event Indication of UNDEFINED message to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>258</b>
<b>Name</b>	Send PPL Event Indication Circuit already Blocked to L5
<b>Description</b>	This function will send a PPL Event Indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>259</b>
<b>Name</b>	Send PPL Event Indication Call Connected to L5
<b>Description</b>	This function will send a PPL Event Indication of Call Connected to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>260</b>
<b>Name</b>	Send PPL Event Indication SND to L5
<b>Description</b>	This function will send a PPL Event Indication of SND to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>261</b>
<b>Name</b>	Form IFAM from Stage N Address Data ICB
<b>Description</b>	This function will format an IFAM TLV message from a Stage N Address Data ICB message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	If successful PPLvINT_EVENT_1 is returned, otherwise PPLvINT_EVENT_0.
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>262</b>
<b>Name</b>	Send PPL Event Indication of Circuit NOT Blocked to L5
<b>Description</b>	This function will format a PPL Event Indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>263</b>
<b>Name</b>	Send PPL Event Indication of Error Timer Expired, Sending CNA to L5
<b>Description</b>	This function will format a PPL Event Indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>264</b>
<b>Name</b>	Send PPL Event Indication to host
<b>Description</b>	This function will format a PPL Event Indication message using the Arg1 value as the event, and send it to the host.
<b>Arg1</b>	The event.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>265</b>
<b>Name</b>	PURGE CHANNEL
<b>Description</b>	This function will cause the Channel to be Purged, with the reason specified in Arg1.
<b>Arg1</b>	The purge reason
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>266</b>
<b>Name</b>	Send PPL Event Indication CFN to L5
<b>Description</b>	This function will send a PPL Event Indication of CFN to L5. In addition if a buffer is present  in the m_message it will be sent also.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>267</b>
<b>Name</b>	Send PPL Event Indication No Response to ACI Request to L5
<b>Description</b>	This function will send a PPL event Indication to the L5 when timer 2 (i.e. TO-14) expires due to not receiving any ACI response from an ACI request.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>268</b>
<b>Name</b>	Send PPL Event Indication of CNG to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>269</b>
<b>Name</b>	Send PPL Event Indication of CNA to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>270</b>
<b>Name</b>	Send PPL Event Indication of RAM to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>271</b>
<b>Name</b>	Send PPL Event Indication of SEM to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>272</b>
<b>Name</b>	Send PPL Event Indication of SAD to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>273</b>
<b>Name</b>	Send PPL Event Indication of SASUI to L5
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>274</b>
<b>Name</b>	Send PPL Event Indication of ASUI to L4
<b>Description</b>	This function will send this indication to L5.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>275</b>
<b>Name</b>	Send PPL Event Indication of No response to SASUI request
<b>Description</b>	This function will send a PPL Event Indication to the L5 when Timer 2 (i.e. TO-14) expires due to not receiving any ASUI response from an SASUI request.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>276</b>
<b>Name</b>	Load Requested ASUI info from Config Bytes
<b>Description</b>	This function will form the appropriate ASUI message into the down working buffer based upon the SASUI request in the up working buffer.  The ASUI message is taken from the Config byte area.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	If it is a valid request PPLevINT_EVENT_1, otherwise PPLevINT_EVENT_0.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>277</b>
<b>Name</b>	Send ASUI to L3
<b>Description</b>	This function will send an ASUI message to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>278</b>
<b>Name</b>	Send SASUI to L3
<b>Description</b>	This function will send a SASUI message to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>279</b>
<b>Name</b>	Send SASUI request message to L3, with user defined type
<b>Description</b>	This function will send a SASUI message to L3, of the type specified in Arg1.
<b>Arg1</b>	The Type of ASUI message to be returned (range 0 to 255). Only 1 is currently valid.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>280</b>
<b>Name</b>	Force sending an ACM message to L3
<b>Description</b>	This function will dynamically build and send an ACM message to L3. The contents of the down working buffer are not accessed or disturbed.
<b>Arg1</b>	
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>281</b>
<b>Name</b>	Set Remote Maintenance Blocking State
<b>Description</b>	This function will allow the user to set the Remote Maintenance Blocking state.
<b>Input Argument 1</b>	0 = Unblocked, 1 = Blocked
<b>Input Argument 2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>282</b>
<b>Name</b>	Set Local Maintenance Blocking State
<b>Description</b>	This function will allow the user to set the Local Maintenance Blocking state.
<b>Input Argument 1</b>	0 = Unblocked, 1 = Blocked
<b>Input Argument 2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>283</b>
<b>Name</b>	Test Blocking State
<b>Description</b>	This function will allow the user to Test any of the following local or remote blocking states. Maintenance, Hardware, and Software.
<b>Input Argument 1</b>	Blocking Location. 0 = Local, 1 = Remote
<b>Input Argument 2</b>	Blocking Type. 0 = Maintenance, 1 = Hardware, 2 = Software.
<b>Outputs</b>	If blocked PPLevINT_EVENT_1 is returned, otherwise PPLevINT_EVENT_0.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>284</b>
<b>Name</b>	Send SAD to L3
<b>Description</b>	This function will send a SAD message to L3.
<b>Input Argument 1</b>	<Not Used>
<b>Input Argument 2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>285</b>
<b>Name</b>	Test Accumulated Digits against Config Byte Digits
<b>Description</b>	<p>This function compares n digits of the accumulated digits string with the digit string contained in the config byte area, at the specified config block and byte number. If the config byte string contains a digit of F that digit is skipped, in the config byte area and the accumulated digit string area.</p> <p><b>NOTE:</b> The digit to begin comparison, within the accumulated digit string is set by the value in register 20.</p>
<b>Arg1</b>	The Config byte block number. Range 1 through 3
<b>Arg2</b>	The byte number within the above block. Range 1 through 200
<b>Outputs</b>	PPLevINT_EVENT_0 if match fails. Or the config bytes exceed the size block limit. PPLevINT_EVENT_1 if match succeeds
<b>Returns</b>	Success always

<b>AF Number</b>	<b>286</b>
<b>Name</b>	Test Accumulated Digits against Immediate Value
<b>Description</b>	<p>This function compares from 1 to 4 digits of the accumulated digits string with the digit string contained in Arg2. If a nibble within Arg2 contains an F that digit is skipped.</p> <p><b>NOTE:</b> The starting digit within the accumulated digit string is set by the value in register 20.</p>
<b>Arg1</b>	The number of bytes present in Arg2 to check. (range 1 to 4)
<b>Arg2</b>	<p>The digits to validate against the accumulated buffer.</p> <p><b>NOTE:</b> A value of 0x4321 would attempt to match a 1 against the first digit in the accumulated digit string, a 2 against the second digit, a 3 against the third...</p>
<b>Outputs</b>	PPLevINT_EVENT_0 if match fails. Or the config bytes exceed the size block limit. PPLevINT_EVENT_1 if match succeeds
<b>Returns</b>	Success always

<b>AF Number</b>	<b>287</b>
<b>Name</b>	Send Alarm Msg to Host
<b>Description</b>	This function will send an ALARM message to the Host using the user defined Severity, Entity, and Alarm values. Please refer to ALARM.H for normal values for these elements.
<b>Arg1</b>	High byte Severity, Low byte Entity
<b>Arg2</b>	Alarm Value
<b>Outputs</b>	Alarm message is sent to L5
<b>Returns</b>	Success always

<b>AF Number</b>	<b>288</b>
<b>Name</b>	Set OOS Reason
<b>Description</b>	This function will Set the Out Of Service Reason to the value specified in Arg1.
<b>Arg1</b>	OOS reason.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	...rw->unique_to.l3p_tup.oos_reason = Arg1
<b>Returns</b>	Success always

<b>AF Number</b>	<b>289</b>
<b>Name</b>	Send Purge Message to L4
<b>Description</b>	Just sends the Purge Message to L4.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>290</b>
<b>Name</b>	Clear Up Working Buffer Ptr
<b>Description</b>	This function will free the buffer attached to the up_working_buffer_ptr, if present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	up_working_buff_ptr is set to NULL.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>291</b>
<b>Name</b>	Set the In-Service_startup_flag to value in Arg1.
<b>Description</b>	This function will set the read write variable "inservice_startup_flag" to the value supplied in Arg1.
<b>Arg1</b>	Value.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	...inservice_startup_flag = value.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>292</b>
<b>Name</b>	Get the In-Service_startup_flag value into register
<b>Description</b>	This function will load the contents of In-Service_startup_flag variable into the register specified in Arg1.
<b>Arg1</b>	Register to contain the value of inservice_startup_flag.
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

## BT IUP CPC (0x0052)

---

**Purpose** The following atomic functions are used by the BT IUP CPC variant

**Atomic Functions** .

<b>AF Number</b>	<b>151</b>
<b>Name</b>	IUPCPC Send IAM to L3P
<b>Description</b>	This function will forward the received IAM message to L3P. L3P will receive it as event IUP_L3PnIAM with an data buffer containing the received IAM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>152</b>
<b>Name</b>	IUPCPC Send IFAM to L3P
<b>Description</b>	This function will forward the received IFAM message to L3P. L3P will receive it as event IUP_L3PnIFAM with an data buffer containing the received IFAM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>153</b>
<b>Name</b>	IUPCPC Send SAM to L3P
<b>Description</b>	This function will forward the received SAM message to L3P. L3P will receive it as event IUP_L3PnSAM with an data buffer containing the received SAM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>154</b>
<b>Name</b>	IUPCPC Send FAM to L3P
<b>Description</b>	This function will forward the received FAM message to L3P. L3P will receive it as event IUP_L3PnFAM with an data buffer containing the received FAM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>155</b>
<b>Name</b>	IUPCPC Send ASUI to L3P
<b>Description</b>	This function will forward the received ASUI message to L3P. L3P will receive it as event IUP_L3PnASUI with an data buffer containing the received ASUI message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>156</b>
<b>Name</b>	IUPCPC Send SND to L3P
<b>Description</b>	This function will forward the received SND message to L3P. L3P will receive it as event IUP_L3PnSND with an data buffer containing the received SND message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>157</b>
<b>Name</b>	IUPCPC Send SAD to L3P
<b>Description</b>	This function will forward the received SAD message to L3P. L3P will receive it as event IUP_L3PnSAD with an data buffer containing the received SAD message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>158</b>
<b>Name</b>	IUPCPC Send SASUI to L3P
<b>Description</b>	This function will forward the received SASUI message to L3P. L3P will receive it as event IUP_L3PnSASUI with an data buffer containing the received SASUI message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>159</b>
<b>Name</b>	IUPCPC Send ACM to L3P
<b>Description</b>	This function will forward the received ACM message to L3P. L3P will receive it as event IUP_L3PnACM with an data buffer containing the received ACM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>160</b>
<b>Name</b>	IUPCPC Send CNG to L3P
<b>Description</b>	This function will forward the received CNG message to L3P. L3P will receive it as event IUP_L3PnCNG with an data buffer containing the received CNG message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success

<b>AF Number</b>	<b>161</b>
<b>Name</b>	IUPCPC Send SEM to L3P
<b>Description</b>	This function will forward the received SEM message to L3P. L3P will receive it as event IUP_L3PnSEM with an data buffer containing the received SEM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>162</b>
<b>Name</b>	IUPCPC Send CNA to L3P
<b>Description</b>	This function will forward the received CNA message to L3P. L3P will receive it as event IUP_L3PnCNA with an data buffer containing the received CNA message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>163</b>
<b>Name</b>	Check Incoming Congestion
<b>Description</b>	This function will check for congestion in the inbound direction.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPLevINT_EVENT_1 if congestion is detected. Otherwise, it is PPLevINT_EVENT_0.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>166</b>
<b>Name</b>	IUPCPC Send ANS to L3P
<b>Description</b>	This function will forward the received ANS message to L3P. L3P will receive it as event IUP_L3PnANS with an data buffer containing the received ANS message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>167</b>
<b>Name</b>	IUPCPC Send CLR to L3P
<b>Description</b>	This function will forward the received CLR message to L3P. L3P will receive it as event IUP_L3PnCLR with an data buffer containing the received CLR message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>168</b>
<b>Name</b>	IUPCPC Send RAM to L3P
<b>Description</b>	This function will forward the received RAM message to L3P. L3P will receive it as event IUP_L3PnRAM with an data buffer containing the received RAM message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>169</b>
<b>Name</b>	IUPCPC Send REL to L3P
<b>Description</b>	This function will forward the received REL message to L3P. L3P will receive it as event IUP_L3PnREL with an data buffer containing the received REL message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>170</b>
<b>Name</b>	IUPCPC Send CCF to L3P
<b>Description</b>	This function will forward the received CCF message to L3P. L3P will receive it as event IUP_L3PnCCF with an data buffer containing the received CCF message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>171</b>
<b>Name</b>	IUPCPC Send BLO to L3P
<b>Description</b>	This function will forward the received BLO message to L3P. L3P will receive it as event IUP_L3PnBLO with an data buffer containing the received BLO message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>172</b>
<b>Name</b>	IUPCPC Send UBL to L3P
<b>Description</b>	This function will forward the received UBL message to L3P. L3P will receive it as event IUP_L3PnUBLwith an data buffer containing the received UBL message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>173</b>
<b>Name</b>	IUPCPC Send BLA to L3P
<b>Description</b>	This function will forward the received BLA message to L3P. L3P will receive it as event IUP_L3PnBLA with an data buffer containing the received BLA message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>174</b>
<b>Name</b>	IUPCPC Send UBA to L3P
<b>Description</b>	This function will forward the received UBA message to L3P. L3P will receive it as event IUP_L3PnUBA with an data buffer containing the received UBA message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>175</b>
<b>Name</b>	IUPCPC Send OVL to L3P
<b>Description</b>	This function will forward the received OVL message to L3P. L3P will receive it as event IUP_L3PnOVL with an data buffer containing the received OVL message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>176</b>
<b>Name</b>	IUPCPC Send CFN to L3P
<b>Description</b>	This function will forward the received CFN message to L3P. L3P will receive it as event IUP_L3PnCFN with an data buffer containing the received CFN message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>178</b>
<b>Name</b>	IUPCPC Send ACI to L3P
<b>Description</b>	This function will forward the received ACI message to L3P. L3P will receive it as event IUP_L3PnACI with an data buffer containing the received ACI message converted into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L3P
<b>Returns</b>	Success always

<b>AF Number</b>	<b>179</b>
<b>Name</b>	Send L5 PPL Event Indication without data.
<b>Description</b>	This function will send a PPL Event Indication to the host with the event number specified in Arg1.
<b>Arg1</b>	The Event Indication Number
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L5
<b>Returns</b>	Success, or the channel will purge if internal errors occur

<b>AF Number</b>	<b>180</b>
<b>Name</b>	IUPCPC Send L5 PPL Event Indication with Raw Data
<b>Description</b>	This function will send a message to L3P indicating that an Invalid ACI message was received from MTP. L3P will receive it as event L3P_L3evINV_ACI with an data buffer containing the received invalid message converted into TLV format.
<b>Arg1</b>	The Event Indication Number
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for L5
<b>Returns</b>	Success always

<b>AF Number</b>	<b>181</b>
<b>Name</b>	Get Timer associated value
<b>Description</b>	Retrieves the associated value
<b>Arg1</b>	Timer number
<b>Arg2</b>	Register to contain associated value
<b>Returns</b>	Success always

<b>AF Number</b>	<b>182</b>
<b>Name</b>	Send PPL Event Indication CLR to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>183</b>
<b>Name</b>	Send PPL Event Indication RE-ANSWER (RAN) to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>184</b>
<b>Name</b>	Send PPL Event Indication SUS to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>185</b>
<b>Name</b>	Send PPL Event Indication RES to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>186</b>
<b>Name</b>	Send PPL Event Indication SWAP to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>187</b>
<b>Name</b>	Send Block Upon Release Event to L3
<b>Arg1</b>	<Not Used>

<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>191</b>
<b>Name</b>	IUPCPC Send IAM to SPRC
<b>Description</b>	This function will convert a TLV IAM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>192</b>
<b>Name</b>	IUPCPC Send IFAM to SPRC
<b>Description</b>	This function will convert a TLV IFAM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>193</b>
<b>Name</b>	IUPCPC Send SAM to SPRC
<b>Description</b>	This function will convert a TLV SAM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>194</b>
<b>Name</b>	IUPCPC Send FAM to SPRC
<b>Description</b>	This function will convert a TLV FAM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>195</b>
<b>Name</b>	IUPCPC Send ASUI to SPRC
<b>Description</b>	This function will convert a TLV ASUI message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>196</b>
<b>Name</b>	IUPCPC Send SND to SPRC
<b>Description</b>	This function will convert a TLV SND message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>197</b>
<b>Name</b>	IUPCPC Send SAD to SPRC
<b>Description</b>	This function will convert a TLV SAD message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>198</b>
<b>Name</b>	IUPCPC Send SASUI to SPRC
<b>Description</b>	This function will convert a TLV SASUI message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>199</b>
<b>Name</b>	IUPCPC Send ACM to SPRC
<b>Description</b>	This function will convert a TLV ACM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>200</b>
<b>Name</b>	IUPCPC Send CNG to SPRC
<b>Description</b>	This function will convert a TLV CNG message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>202</b>
<b>Name</b>	IUPCPC Send CNA to SPRC
<b>Description</b>	This function will convert a TLV CNA message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>203</b>
<b>Name</b>	IUPCPC Send SEM to SPRC
<b>Description</b>	This function will convert a TLV SEM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>206</b>
<b>Name</b>	IUPCPC Send ANS to SPRC
<b>Description</b>	This function will convert a TLV ANS message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>207</b>
<b>Name</b>	IUPCPC Send CLR to SPRC
<b>Description</b>	This function will convert a TLV CLR message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>208</b>
<b>Name</b>	IUPCPC Send RAM to SPRC
<b>Description</b>	This function will convert a TLV RAM message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>209</b>
<b>Name</b>	IUPCPC Send REL to SPRC
<b>Description</b>	This function will convert a TLV REL message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>210</b>
<b>Name</b>	IUPCPC Send CCF to SPRC
<b>Description</b>	This function will convert a TLV CCF message into raw format and queue it up for SPRC or MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>211</b>
<b>Name</b>	IUPCPC Send BLO to SPRC
<b>Description</b>	This function will convert a TLV BLO message into raw format and queue it up for SPRC or MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>212</b>
<b>Name</b>	IUPCPC Send UBL to SPRC
<b>Description</b>	This function will convert a TLV UBL message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>213</b>
<b>Name</b>	IUPCPC Send BLA to SPRC
<b>Description</b>	This function will convert a TLV BLA message into raw format and queue it up for SPRC or MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>214</b>
<b>Name</b>	IUPCPC Send UBA to SPRC
<b>Description</b>	This function will convert a TLV UBA message into raw format and queue it up for SPRC or MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>215</b>
<b>Name</b>	IUPCPC Send OVL to SPRC
<b>Description</b>	This function will convert a TLV OVL message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>216</b>
<b>Name</b>	IUPCPC Send CFN to SPRC
<b>Description</b>	This function will convert a TLV CFN message into raw format and queue it up for SPRC or MTP.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>218</b>
<b>Name</b>	IUPCPC Send ACI to SPRC
<b>Description</b>	This function will convert a TLV ACI message into raw format and queue it up for SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for SPRC
<b>Returns</b>	Success always

<b>AF Number</b>	<b>219</b>
<b>Name</b>	IUPCPC SEND REL w/data to SPRC
<b>Description</b>	This function will format a raw Release message with the reason suPPLied in Arg1 and queue it up for MTP
<b>Arg1</b>	Decimal value of Release Reason
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	Message queued for MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>220</b>
<b>Name</b>	IUPCPC Send CNA w/data to SPRC
<b>Description</b>	This function will format a raw Connection Not Admitted message with the reason suPPLied in Arg1 and queue it up for SPRC.
<b>Arg1</b>	Decimal value of CNA Reason
<b>Arg2</b>	Decimal value of CNA Diagnostics
<b>Outputs</b>	Message queued for SPRC or MTP
<b>Returns</b>	Success or PPLerrAF_INITIATED_RESET if the send to MTP fails

<b>AF Number</b>	<b>221</b>
<b>Name</b>	Check for Incoming Congestion
<b>Description</b>	This function will check an internal variable to see if incoming congestion has been detected.
<b>Arg1</b>	NA
<b>Arg2</b>	NA
<b>Outputs</b>	PPLevINT_EVENT_0 when no congestion exists. PPLevINT_EVENT_1 when Congestion IS Present.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>232</b>
<b>Name</b>	IUPCPC Test if Circuit is Blocked
<b>Description</b>	This function will test the blocked state of the circuit. The user may specify which of three types of blocking conditions for both local and remote circuit ends. The routine will return with a PPL event corresponding to it's state. PPLevINT_EVENT_0 if not blocked, or PPLevINT_EVENT_1 if blocked.
<b>Arg1</b>	Circuit end to check. 0 = Local, 1 = Remote
<b>Arg2</b>	Type of Block to check. 0 = Maintenance, 1 = Hardware, 2 = Software
<b>Outputs</b>	See above
<b>Returns</b>	Success always

<b>AF Number</b>	<b>233</b>
<b>Name</b>	IUPCPC Set Local Circuit Maintenance status to user specified state.
<b>Description</b>	This function will set the local circuit's maintenance status to the value specified in Arg1.
<b>Arg1</b>	State
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>236</b>
<b>Name</b>	IUPCPC Validate and Convert SPRC message to TLV format
<b>Description</b>	This function will validate and convert raw message into TLV format.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	A buffer containing the TLV format of the raw message.  If successful PPLvINT_EVENT_1, otherwise PPLvINT_EVENT_0.
<b>Returns</b>	Success always

<b>AF Number</b>	<b>247</b>
<b>Name</b>	IUPCPC Send Unknown MSG to SPRC
<b>Description</b>	This function will send a message containing the Generic TLV Entry Data to raw format, and send it to SPRC.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success always

<b>AF Number</b>	<b>250</b>
<b>Name</b>	IUPCPC Get IUP message Parameter byte from the TLV into GPR #19
<b>Description</b>	This function will search for the TLV ID and byte specified in args 1 and 2 and return with GPR 19 containing the value of the specified byte.
<b>Arg1</b>	The TLV ID to be retrieved.
<b>Arg2</b>	The byte within the TLV to be place into GPR 19
<b>Outputs</b>	If successful GPR 19 will contain the specified byte and return with PPLvINT_EVENT_1.  Otherwise, GPR 19 will be set to 0 and we return PPLvINT_EVENT_0.
<b>Returns</b>	Success

<b>AF Number</b>	<b>253</b>
<b>Name</b>	IUPCPC Update Remote Circuit Maintenance Block State to users specified state.
<b>Description</b>	This function will set an internal variable indicating the state of Remote Circuit's maintenance block state.
<b>Arg1</b>	The State. 0 = Unblocked, 1 = Blocked
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	<Not Used>
<b>Returns</b>	Success

## SCCP SCRC (0x0066)

---

**Purpose** The following atomic functions are used by the PPL component, SCRC (SCCP Routing Control - 0x0066).

**Atomic Functions** .

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Test whether Global Title Translation (GTT) is allowed for local sub-system.
<b>Description</b>	Tests whether GTT is needed and the sub-system number (SSN) is allowed to do GTT locally.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLevINT_EVENT_1 if yes.  PPLevINT_EVENT_0 if GTT is not needed

<b>AF Number</b>	<b>93</b>
<b>Name</b>	SCRC Global Title Translation
<b>Description</b>	This function performs GTT, returns the success or failure status, and determines whether to perform further treatment to the address after the GTT.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_0: success  PPLvINT_EVENT_1: No translation for an address of such nature  PPLvINT_EVENT_2: No translation for this specific address  PPLvINT_EVENT_3: SSN failure  PPLvINT_EVENT_4: SCCP unequipped  PPLvINT_EVENT_5: MTP failure  PPLvINT_EVENT_6: MSG overlength with new CDPA  PPLvINT_EVENT_7: Support UMTS calling party address treatment.

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Universal mobile telecommunications systems (UMTS) calling party address (CGPA) treatment
<b>Description</b>	Supports UMTS CGPA treatment.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Result</b>	PPLvINT_EVENT_0 if success  PPLvINT_EVENT_6 if fail

<b>AF Number</b>	<b>95</b>
<b>Name</b>	SCRC updates the called party address (CDPA).
<b>Description</b>	Updates CDPA in the internal chained buffer with new CDPA results from GTT.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Result</b>	Void

# 7 ISDN Atomic Functions

**Purpose** This chapter documents the atomic functions associated with the ISDN PPL components that are released for customization using the PPL Composer. Some of the ISDN PPL components are used with the V5.2 interface.

## L3P Call Control (0x0005) Atomic Functions

---

**Purpose** This section describes the atomic functions for the L3P Call Control PPL component.

**L3P Call Control Atomic Functions** The following atomic functions are specific to the ISDN L3P Call Control PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Load Raw IE in L4 Buffer
<b>Description</b>	Loads a raw IE received from the distant end into a Raw IE ICB (0x11) to be sent to the host in a <i>PPL Event Indication</i> or <i>PPL Event Request</i> message for service with address data. If the IE is not received, then nothing is loaded.
<b>Arg1</b>	<IE Value> 0-255 (ITU-T Q.931 codeset 0)
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Load Formatted and Raw IEs in L4 Buffer
<b>Description</b>	Loads the formatted IE ICB subtype (0x10) and the raw IE type (0x11) with the IEs received. The supported formatted IEs can be loaded and all other IES are included as raw in the Raw ICB. Eventually, you must invoke an atomic function to send this information.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Copy Raw IEs into L4 Buffer
<b>Description</b>	Copies received IEs into the Raw IE ICB subtype (0x11) for sending at a later time.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Purge B Channel and Send Clear to L3
<b>Description</b>	Deletes all timers and frees DSP resources per B channel. Then, it releases the call to Layer 3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Load Formatted IEs into L4 Buffer
<b>Description</b>	Loads specific formatted IEs into a formatted IE subtype (0x10) for later reporting to the host.
<b>Arg1</b>	<Formatted IE Type> 1-50
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Test if IE Exists in L4 Message
<b>Description</b>	Tests for an existing IE in a Layer 4 message for <i>PPL Event Request</i> and <i>Outseize Control</i> .
<b>Arg1</b>	<IE Type> 0-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send BSM CR Purge Indication
<b>Description</b>	Sends the BSM (component 7) a purge indication that should return the In-Service DS0 Status Change to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send Purge Indication to L4
<b>Description</b>	Sends an L3P initiated purge to Layer 4.
<b>Arg1</b>	<Purge Value> 0-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Load L4 IEs in Sorted Order
<b>Description</b>	Translates formatted and raw IEs to be sent to Layer 3 in sorted order.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Load L4 Formatted IEs to L3 Buffer
<b>Description</b>	Loads just the formatted IEs into a buffer to Layer 3. These would be inserted by host in the <i>PPL Event Request</i> and the <i>Outseize Control</i> .
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Load L4 Raw IEs to L3 Buffer
<b>Description</b>	Loads the Layer 4 Raw IEs into a buffer to send to Layer 3 at a later time.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>65</b>
<b>Name</b>	PPL Send Host <i>PPL Event Indication</i>

<b>Description</b>	Sends a <i>PPL Event Indication</i> to host. If a previous atomic function's loaded Layer 3 IEs to go to Layer 4, then they also get sent.
<b>Arg1</b>	<Event ID> 1-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Test L3 Error Indication and Map to Internal Event
<b>Description</b>	If the previous event was a Layer 3 Error Indication, the error is added to the internal event base and returned. The state that follows this error should be an internal state to test the value.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Test # of Stored Digits
<b>Description</b>	Test if the # of stored digits is less than or equal to the value in Arg1.
<b>Arg1</b>	<# of Digits> See AF 84
<b>Arg2</b>	<Not Used>

<b>Test Result</b>	<b>PPL Internal Event</b>
<=	1
Other	0

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Test if IEs Exist in L3 Message
<b>Description</b>	Tests to see if the IE indicated by Arg1 exists in a Layer 3 message.
<b>Arg1</b>	<IE ID> 0-255 (ITU-T Q.931 codeset 0 IE value)
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Exists	1
Does not exist	0

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Load IE Library IE
<b>Description</b>	Copies previously stored IEs in the IE Library into the buffer to Layer 3.
<b>Arg1</b>	<IE Library Index> 0-30
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Test if Stored Digits Equals Config Byte Value
<b>Description</b>	Tests to see if # of stored digits is less than or equal the value in Arg1.
<b>Arg1</b>	<Config Byte>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
<=	1
Other	0

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Test For Sending Complete Indication
<b>Description</b>	Tests to see if there is a Sending Complete Indication message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Yes	1
No	0

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Load Cause Code IE
<b>Description</b>	Copies the Cause Code indicated in Arg1 into a buffer to send to L3.
<b>Arg1</b>	<Cause Code>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Test First Data ICB in <i>Outseize Control</i> Message
<b>Description</b>	Tests the first Data ICB subtype in the <i>Outseize Control</i> message. This lets the Layer 3 Plus protocol distinguish the BCD encoded format for sending the information, and the formatted and raw subtypes. The returned code is the internal event for the value.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Translate Network-specific IE from B Channel Option
<b>Description</b>	Looks into channel previously configured options ( <i>B Channel Configure</i> ) and creates and copies the network-specific IE in the buffer to Layer 3. This atomic function is not needed if the host includes this information in the <i>Outseize Control</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Translate Bearer Capability IE from B Channel Option
<b>Description</b>	Looks into channel previously configured options ( <i>B Channel Configure</i> ) and creates and copies the Bearer Capability IE in the buffer to Layer 3. This atomic function is not needed if the host includes this information in the <i>Outseize Control</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Load Outseize Control Address Digits to Q.931 IEs
<b>Description</b>	Takes the BCD encoded stage digits from the <i>Outseize Control</i> and creates the Called Party IE and loads it in a buffer to send to Layer 3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Send <i>Outseize Control</i> Ack
<b>Description</b>	Sends an <i>Outseize Control</i> ACK.
<b>Arg1</b>	<Value> 0-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send <i>PPL Event Request</i> Ack
<b>Description</b>	Sends a <i>PPL Event Request</i> ACK.
<b>Arg1</b>	<Value> 0-255
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send L3 to L4 SETUP Indication Using BCD-Encoded Stages
<b>Description</b>	Extracts Layer 3 SETUP information to load called and calling party digits in the <i>Request for Service with Address Data</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send L3 to L4 SETUP Indication Exact L3 Frame
<b>Description</b>	Sends the exact received frame for the SETUP message to the host in the <i>Request for Service with Data</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Test Bit Mask in Specified Byte Offset Of IE in L2 to L3 Message
<b>Description</b>	Tests any set of bits in any byte of an IE in a message from L3.
<b>Arg1</b>	<IE ID>
<b>Arg2</b>	<Byte Offset/Bit Mask> Byte Offset (MSB)/Bit Mask (LSB)

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Test for Sufficient Digits In Stored Called Party IE Buffer
<b>Description</b>	Verifies that there are sufficient digits in the Called Party IE Buffer. The number of significant digits is indicated in Config Byte 10. <b>NOTE:</b> This atomic function applies to Euro ISDN only
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Store Called Party IE Digits
<b>Description</b>	Stores the Called Party digits from the Called Party IE in a SETUP or INFORMATION message from L3 into the Overlap Receive buffer. See AF 68 and AF 71.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Send INFO Requests in Overlap Mode
<b>Description</b>	Sends Called Party digits in continuous INFORMATION messages to L3 with one Called Party digit per message. This function should only be used for testing. The Call Type must be set to 1 (Called Party digits are in Call Type ICB) using the Call Type ICB in the <i>Outseize Control</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Test Call Type
<b>Description</b>	Tests the Call Type obtained from the Call Type ICB sent in an <i>Outseize Control</i> message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Call Type 1 Called Party digits are in Call Type ICB	1
Call Type 2 Host controls sending of INFORMATION messages 2 Call Type ICB does not exist	0

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Load Sending Complete IE
<b>Description</b>	Load Sending Complete IE in to a SETUP or INFORMATION message to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Test Called Party Digits using Config Byte
<b>Description</b>	Tests 1 or 2 digits in the Stored Called Party buffer. The offset into the digit string is indicated in Arg1. The number of digits to test (1 or 2) is indicated in Arg2.
<b>Arg1</b>	<Config Byte Offset> Config Byte containing the offset in to the digit string.
<b>Arg2</b>	<Config Byte # Test Digits> 1 or 2

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Test Called Party Digits
<b>Description</b>	Tests 1 or 2 digits in the Stored Called Party buffer. The offset into the digit string is indicated in Arg1. The number of digits to test (1 or 2) is indicated in Arg2.
<b>Arg1</b>	<Offset> Offset in to the digit string.
<b>Arg2</b>	<# of Test Digits> 1 or 2

<b>AF Number</b>	<b>91</b>
<b>AF Name</b>	Load Remaining Called Party Digits in L3L4 Formatted IE Buffer
<b>AF Description</b>	Load remaining Called Party digits received in INFORMATION message from L3 into Formatted IE buffer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Store L3L4 Form and Raw IEs
<b>Description</b>	Store the L3 to L4 Formatted and Raw IE buffers to be loaded later.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Load Stored L3L4 Form and Raw IE Buffer
<b>Description</b>	Load the stored Formatted and Raw IE buffers.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Send L3 ANI Request
<b>Description</b>	Sends an ANI feature request to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Send L3 Vari-A-Bill Request
<b>Description</b>	Sends a Vari-a-Bill feature request to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Load NSF IE with ANI Req Information from B Channel Option
<b>Description</b>	Load a Network-specific Facility IE with ANI on Demand request information from the B channel configuration settings.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Decode NSF IEs to Set Available Features And Services
<b>Description</b>	Decodes the Network-specific Facility IE in the incoming SETUP message to setup the available features and services supported by the call.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Test if Feature is Available
<b>Description</b>	Test if the feature indicated by Arg1 is supported for this call, as indicated in the Network-specific Facility IE in the SETUP message.
<b>Arg1</b>	<Feature #> See TR41459, Appendix 3, Section 4.2.2.2.
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Supported	0
Not Supported	1

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Test if Service is Supported
<b>Description</b>	Test if the service indicated by Arg1 is supported for this call, as indicated in the Network-specific Facility IE in the SETUP message.
<b>Arg1</b>	<Service #> See TR41459, Part 3, Section 3.6.5.19.
<b>Arg2</b>	<Not Used>

Test Result	PPL Internal Event
Supported	0
Not Supported	1

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Send L3 SETUP Request
<b>Description</b>	Request Layer 3 to send a SETUP message. This includes any previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Send L3 CALL PROCEEDING Request
<b>Description</b>	Requests Layer 3 to send a CALL PROCEEDING message. This includes any previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Send L3 INFO Request
<b>Description</b>	Requests Layer 3 to send an INFORMATION message. This includes all previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Send L3 ALERTING Request
<b>Description</b>	Requests Layer 3 to send an ALERTING message. This includes all previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Send L3 PROGRESS Request
<b>Description</b>	Requests Layer 3 to send a PROGRESS message. This includes all previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Send L3 CONNECT Request
<b>Description</b>	Requests Layer 3 to send a CONNECT message. This includes all previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Send L3 CLEAR Request
<b>Description</b>	Requests Layer 3 to send a DISCONNECT or a RELEASE message. This includes an previously stored IEs to be sent. The Clear Request representation is determined by the side that initiated the DISCONNECT.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Send L3 USER INFORMATION Request
<b>Description</b>	Requests Layer 3 to send a USER INFORAMTION message. This includes any previously saved IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>108</b>
<b>Name</b>	Send L3 FACILITY Request
<b>Description</b>	Request Layer 3 to send a FACILITY message. This includes previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>109</b>
<b>Name</b>	Send L3 SEGMENT Request
<b>Description</b>	Send SEGMENT REQ message to L3.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>111</b>
<b>Name</b>	Send L3 REGISTER Request
<b>Description</b>	Requests Layer 3 to send a REGISTER message. This includes previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>112</b>
<b>Name</b>	Store Call Type ICB
<b>Description</b>	Stores the Call Type ICB if one exists.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>113</b>
<b>Name</b>	Send L3 a GENERIC EVENT Request
<b>Description</b>	Sends a GENERIC EVENT request to L3.
<b>Arg1</b>	<Event #>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>116</b>
<b>Name</b>	Send L3 NOTIFY Request
<b>Description</b>	Requests Layer 3 to send a NOTIFY message including any previously stored IEs to be sent.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>117</b>
<b>Name</b>	Send L3 MORE INFO Request
<b>Description</b>	Sends a MORE INFO request to L3 to initiate Overlap Sending.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Send L4 Setup Indication
<b>Description</b>	Informs Layer 4 of an incoming call; sends the Request For Service with Address Data to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Send L4 ALERTING Indication (Connect TS Post-answer)
<b>Description</b>	Sends an alerting indication to Layer 4, which causes a two-way connection if the host sent a Connect (A, B) message previously. Do not use this form to make a two-way connection prior to answer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>133</b>
<b>Name</b>	Send L4 ALERTING Indication (Connect TS Now)
<b>Description</b>	Sends the alerting indication to Layer 4 but not until a connection answer occurs.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Send L4 CONNECT Indication
<b>Description</b>	Informs Layer 4 of distant end answer.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Send L4 DISCONNECT Indication
<b>Description</b>	Inform Layer 4 of a DISCONNECT request from the distant end.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Send L4 CLEAR Indication
<b>Description</b>	Informs Layer 4 that the release sequence is complete and the call has returned to IDLE/NULL State.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>137</b>
<b>Name</b>	Determine Congestion Level
<b>Description</b>	Finds out the current congestion level
<b>Arg1</b>	N/A
<b>Arg2</b>	N/A

<b>AF Number</b>	<b>138</b>
<b>Name</b>	Send L4 Access Denied
<b>Description</b>	Sends a L3_CCPnACCESS_DENIED message to Layer 4 with status ISDN_STACK_CONGESTED (0x3E)
<b>Arg1</b>	Value of status field
<b>Arg2</b>	N/A

<b>AF Number</b>	<b>144</b>
<b>Name</b>	Save type of Segmented Message in GPR if REASSEMBLY< GPR#>
<b>Description</b>	Saves Segmented Message type in GPR specified by Arg1, if REASSEMBLY flag is set in message
<b>Arg1</b>	GPR#
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>145</b>
<b>Name</b>	Save type of Segmented Message in GPR if REASSEMBLY and Release Request Enabled< GPR#>
<b>Description</b>	Saves Segmented Message type in GPR specified by Arg1, if REASSEMBLY flag is set in message and Channel Release Request is enabled on ISDN disconnect
<b>Arg1</b>	GPR#
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>146</b>
<b>Name</b>	Compare type of Segmented Message with <GPR#>
<b>Description</b>	Compare type of Segmented Message with value of GPR specified by Arg1
<b>Arg1</b>	GPR#
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINTERNAL_EVENT_1 if Segmented Message type == GPR <Arg1> PPLvINTERNAL_EVENT_0 if Segmented Message type != GPR <Arg1>

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Send L4 Segment Indication
<b>Description</b>	Send to SEGMENT Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>148</b>
<b>Name</b>	Test for last segment of Segmented message
<b>Description</b>	Tests for last segment of Segmented message.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINTERNAL_EVENT_1 if last Segment PPLvINTERNAL_EVENT_0 if not last Segment

<b>AF Number</b>	<b>149</b>
<b>Name</b>	Clear LP3_CC IE buffers
<b>Description</b>	Sets LP3_CC IE buffer pointers to NULL
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

## L3P B Channel Control (0x0007) Atomic Functions

---

**Purpose** This section describes the atomic functions used by the L3 Call Reference PPL component.

- 

**L3P B Channel Control atomic functions** The following AFs are specific to the ISDN L3P B Channel Control PPL component. See *Common Atomic Functions (2-1)* for the AFs common to all PPL components.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send Enable B Channel to L3
<b>Description</b>	Enables a B channel to Layer 3 and forces the SERVICE of in-service to be sent to the distant end.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Send Disable B Channel to L3
<b>Description</b>	Enables a B channel to Layer 3 and forces the service of Out-of-Service to be sent to the distant end.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Send L3 B Channel Maintenance to L3
<b>Description</b>	Enables a B channel to Layer 3 and forces a service of maintenance to be sent to the distant end.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send CCSM Service State Event
<b>Description</b>	Forces an indication to Layer 4 and causes a <i>DS0 Status Change</i> message to be sent to the host.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Test to see if the B channel is on the D channel Facility
<b>Description</b>	Checks to see if the B channel for this PPL resides on the same span as any other D channel (primary or backup).
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send Remove Call to L3P CR
<b>Description</b>	Remove Call is sent to L3P CR
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

## L3 Call Reference (0x0008) Atomic Function

---

**Purpose** This section describes the atomic functions used by the L3 Call Reference PPL component.

**L3 Call Reference Atomic Function** The following Atomic Function supports congestion control functionality to the ISDN PRI card. See *Common Atomic Functions (2-1)* for the AFs common to all PPL components.

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Determine congestion level
<b>Description</b>	Finds out the current congestion level
<b>Arg1</b>	N/A
<b>Arg2</b>	N/A

The following Atomic Functions support QSIG/PSS1 Basic Call Signaling functionality. See *Common Atomic Functions (2-1)* for the AFs common to all PPL components.

<b>AF Number</b>	<b>140</b>
<b>Name</b>	Get Segmentation Event from L3 DSM message
<b>Description</b>	Extracts event coded message from L3DSM and returns corresponding internal event
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLevINTERNAL_EVENT_0 and Segmentation Event from L3 DSM

<b>AF Number</b>	<b>141</b>
<b>Name</b>	Send SEGMENT Indication to L3P
<b>Description</b>	Send SEGMENT_IND message to LP3
<b>Arg1</b>	Event (1:3)
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>142</b>
<b>Name</b>	Send SEGMENT Request to L3DSM
<b>Description</b>	Send SEGMENT_REQ message to L3DSM
<b>Arg1</b>	Event (1:3)
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>143</b>
<b>Name</b>	Validate called party number IE
<b>Description</b>	Validates called party number IE
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if IE is valid PPLvINT_EVENT_0 if IE is missing PPLvINT_EVENT_2 if IE is invalid

<b>AF Number</b>	<b>144</b>
<b>Name</b>	Validate call state IE
<b>Description</b>	Checks if the call state indicated in the IE matches the current state. Validation is done if the IE is present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if IE is valid PPLvINT_EVENT_0 if IE is missing or call state is invalid

<b>AF Number</b>	<b>145</b>
<b>Name</b>	Validate cause IE
<b>Description</b>	Validates cause IE. Validation is done if the IE is present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if IE is valid PPLvINT_EVENT_0 if IE is missing PPLvINT_EVENT_2 if IE is invalid

<b>AF Number</b>	<b>146</b>
<b>Name</b>	Validate progress indicator IE
<b>Description</b>	Validates progress indicator IE. Validation is done if the IE is present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if IE is valid PPLvINT_EVENT_0 if IE is missing PPLvINT_EVENT_2 if IE is invalid

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Validate channel IE
<b>Description</b>	Validates channel IE. Validation is done if the IE is present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if IE is valid PPLvINT_EVENT_0 if IE is missing PPLvINT_EVENT_2 if IE is invalid

<b>AF Number</b>	<b>148</b>
<b>Name</b>	Validate bearer capability IE
<b>Description</b>	Validates bearer capability IE. Validation is done if the IE is present.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLevINT_EVENT_1 if IE is valid PPLevINT_EVENT_0 if IE is missing PPLevINT_EVENT_2 if IE is invalid

## L3 Global Call Reference (0x0009) Atomic Function

---

**Purpose** This section describes the atomic functions and PPL events for the L3 Global Call Reference PPL component.

**L3 Global Call Reference PPL Atomic Functions** The following Atomic Function in Layer 3 GCR PPL Component (0x0009) adds functionality to provide for ISDN Bearer Connection Independent Supplementary Services. See *Common Atomic Functions (2-1)* for the AFs common to all PPL components.

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Setup Call Reference
<b>Type</b>	Normal
<b>Description</b>	Sets up the Layer 3 GCR call reference data structure using the call reference specified in the ICB.
<b>Arg1</b>	N/A
<b>Arg2</b>	N/A

### Results of Atomic Function 88

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Normal
<b>Type</b>	Normal /Blocking/Test
<b>Description</b>	Normal /Blocking/Test
<b>Arg1</b>	Description
	Input Range
<b>Arg2</b>	Description
	Input Range

**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 D Channel Control (0x000A) Atomic Functions

---

**Purpose** This section includes the atomic functions and events for the ISDN PPL component, L3 D Channel Control.

**L3 D Channel Control Atomic Functions** The following Atomic Function supports congestion control functionality to the ISDN PRI card. See Common Atomic Functions in the Chapter 2 for the AFs common to all PPL components.

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Test Congestion Control Thresholds
<b>Description</b>	Tests for congestion. If call averages or burst have been exceeded alarm messages are sent to the host.
<b>Arg1</b>	N/A
<b>Arg2</b>	N/A

The following Atomic Functions support QSIG/PSS1 Basic Call Signaling functionality. See *Common Atomic Functions (2-1)* for the AFs common to all PPL components.

<b>AF Number</b>	<b>74</b>
<b>Name</b>	REASSEMBLY: Validate First Segment
<b>Description</b>	Verifies Segment Message is received in the Reassembly Null state
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if valid Segment Message PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>75</b>
<b>Name</b>	REASSEMBLY: Respond to Invalid First Segment
<b>Description</b>	Depending on call state and configuration byte, responds with STATUS/STATUS_ENQUIRY or RELEASE_COMPLETE message to invalid First Segment
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>76</b>
<b>Name</b>	REASSEMBLY: Test Message Type
<b>Description</b>	Checks whether the message type is SEGMENT or not
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if message type is SEGMENT PPLvINT_EVENT_0 if message type is not SEGMENT

<b>AF Number</b>	<b>77</b>
<b>Name</b>	REASSEMBLY: Validate Subsequent Segment
<b>Description</b>	Verifies Segment Message received in Reassembly Active state
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if valid Segmented Message PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>78</b>
<b>Name</b>	REASSEMBLY: Store Call Request, Segment Message type, number of segments remaining
<b>Description</b>	Stores important data about the current reassembly process
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>79</b>
<b>Name</b>	REASSEMBLY: Accumulate segment in buffer
<b>Description</b>	Adds the segment to the reassembly buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>82</b>
<b>Name</b>	REASSEMBLY: Test number of segments remaining
<b>Description</b>	Checks if there are more segments to come
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLevINT_EVENT_1 if this is last segment PPLevINT_EVENT_0 if more segments are expected

<b>AF Number</b>	<b>83</b>
<b>Name</b>	REASSEMBLY: Store number of segments remaining
<b>Description</b>	Stores number of remaining segments
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>84</b>
<b>Name</b>	REASSEMBLY: Assemble, reorder and send
<b>Description</b>	Reorders the IEs in the buffer, so that important IEs come first. It then sends the IEs to L3 Call Reference, first as a normal call processing message with a Segmented Message IE, followed by segmented messages.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>85</b>
<b>Name</b>	REASSEMBLY: Send to L3 Call Reference
<b>Description</b>	Sends the message to L3 Call Reference
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>86</b>
<b>Name</b>	REASSEMBLY: Discard accumulated segments
<b>Description</b>	Discards the segments in the reassembly buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>87</b>
<b>Name</b>	SEGMENTATION: Test for valid SEGMENT message
<b>Description</b>	Checks whether message is a valid SEGMENT message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if message is a valid SEGMENT message PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>88</b>
<b>Name</b>	SEGMENTATION: Test for Segmentation Data in message
<b>Description</b>	Checks whether the message contains Segmentation Data
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if Segmentation Data is present PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>89</b>
<b>Name</b>	SEGMENTATION: Verify Segmentation Data
<b>Description</b>	Tests the validity of Segmentation Data
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if Segmentation Data is valid PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>90</b>
<b>Name</b>	SEGMENTATION: Store Call Request, Segment Message type, number of segments remaining, add to buffer
<b>Description</b>	Allocates the memory buffer for this segmentation, stores important data, converts call processing message to Segment Message, adds to buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>91</b>
<b>Name</b>	SEGMENTATION: Store number of remaining segments, add to buffer
<b>Description</b>	Stores the new number of segments remaining, adds the segment to buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>92</b>
<b>Name</b>	SEGMENTATION: Discard segments
<b>Description</b>	Frees the allocated buffer and discards the received segments
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>93</b>
<b>Name</b>	SEGMENTATION: Free segmentation buffer for all channels
<b>Description</b>	Frees the allocated buffer and discards the received segments for all channels
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>94</b>
<b>Name</b>	SEGMENTATION: Test number of segments remaining
<b>Description</b>	Checks if there are more segments to come
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if this is last segment PPLvINT_EVENT_0 if more segments are expected

<b>AF Number</b>	<b>95</b>
<b>Name</b>	SEGMENTATION: Set segmentation state
<b>Description</b>	Sets the segmentation state for the current B channel
<b>Arg1</b>	State (0/1)
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>96</b>
<b>Name</b>	SEGMENTATION: Send accumulated segments and discard them
<b>Description</b>	Sends the segments in the buffer to L2 and frees the buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>

<b>AF Number</b>	<b>97</b>
<b>Name</b>	SEGMENTATION: Test segmentation state
<b>Description</b>	Tests the segmentation state for the current B channel
<b>Arg1</b>	State (0/1)
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if Arg1 is equal to the segmentation state PPLvINT_EVENT_0 if otherwise

<b>AF Number</b>	<b>98</b>
<b>Name</b>	SEGMENTATION: Test Segment Message type and Call Reference
<b>Description</b>	Tests the Segment Message type for validity
<b>Arg1</b>	State (0/1)
<b>Arg2</b>	<Not Used>
<b>Results</b>	PPLvINT_EVENT_1 if message is not a SEGMENT message and has a different Call Reference from the current reassembly PPLvINT_EVENT_0 if message is a SEGMENT message or not a SEGMENT message, but has the same Call Reference from the current reassembly

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Send DSM Segmentation Event to L3 Call Reference
<b>Description</b>	Send Segmentation Event Indication through L3 Call Reference to host
<b>Arg1</b>	Event (1:3)
<b>Arg2</b>	<Not Used>

### 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 *
* Applies to D channel Backup only.	

## L3P PSTN (0x0091) Atomic Functions

---

**Purpose** This section describes the atomic functions for the L3P PSTN PPL component.

**L3P PSTN Atomic Functions** The following atomic functions are specific to the ISDN L3P PSTN PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send L3 ESTABLISH Request using L3 Outgoing Buffer
<b>Description</b>	This function sends an Establish Request to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Send L3 ESTABLISH Acknowledge Request using L3 Outgoing Buffer
<b>Description</b>	This function sends an Establish Acknowledge Request to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Send L3 SIGNAL Request using L3 Outgoing Buffer
<b>Description</b>	This function sends a Signal message to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send L3 STATUS Request using L3 Outgoing Buffer
<b>Description</b>	This function sends a Status message to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send L3 DISCONNECT Request using L3 Outgoing Buffer
<b>Description</b>	This function sends a Disconnect message to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send L3 DISCONNECT COMPLETE Request using L3 Outgoing Buffer
<b>Description</b>	This function sends a Disconnect Complete Request to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Send L3 PROTOCOL PARAMETER Request using L3 Outgoing Buffer
<b>Description</b>	This function sends a Protocol Parameter Request to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Send L3 signal messages with stored called party digits from L4
<b>Description</b>	Sends signal messages to L3, each with a digit signal IE from called party digits from L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Send L4 SETUP Indication (Formatted IEs ICB) using L4 Outgoing Buffer
<b>Description</b>	This function sends a SETUP Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send L4 ALERTING Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a ALERTING Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send L4 CONNECT Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a CONNECT Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send L4 CUT Through Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a Cut Through Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send L4 PROGRESS Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a Progress Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send L4 DISCONNECT Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a Disconnect Indication to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send L4 CLEAR Request using L4 Outgoing Buffer
<b>Description</b>	This function sends a Clear Request message to L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>67</b>
<b>Name</b>	Send Access Denied message with call control flag
<b>Description</b>	Outseize Nack and Clear indication will be sent from L4
<b>Arg1</b>	Mentions Status
<b>Arg2</b>	Mentions More Status in the event Arg1 value is 0x51
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>69</b>
<b>Name</b>	Send L4 PURGE Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends a Purge Indication to L4
<b>Arg1</b>	Purge Reason
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Send L4 Outseize Acknowledge Indication using L4 Outgoing Buffer
<b>Description</b>	This function sends an L4 Outseize Acknowledge Indication to L4
<b>Arg1</b>	Status
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send L4 Channel Status message
<b>Description</b>	Sends a Channel Status message to L4
<b>Arg1</b>	Status (0 = Out of Service/1 = In Service)
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send L4 RTR Route Control Acknowledge <status>
<b>Description</b>	Sends a Route Control Acknowledge to L4 Router Status given in Arg1
<b>Arg1</b>	Status
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send L4 RTR Route Control Reject <status>
<b>Description</b>	Sends a Route Control Reject to L4 Router
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send BCC Channel Request
<b>Description</b>	This function sends a Channel (timeslot) Request message to L3P BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send BCC Channel Remove
<b>Description</b>	This function sends a Channel (timeslot) Remove message to L3P BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Send BCC Channel Indication
<b>Description</b>	This function sends a Channel (timeslot) Indication message to L3P BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Send BCC Channel Error Indication
<b>Description</b>	This function sends an Channel Error Indication message to L3P BCC
<b>Arg1</b>	Error Code
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>79</b>
<b>Name</b>	Send BCC Deferred Blocking Complete
<b>Description</b>	This function sends a Deferred Blocking Complete message to L3P BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send PPL Event Indication to L5 <Event>
<b>Description</b>	Sends a PPL Event to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send PPL Event Request Acknowledge to L5 <Status>
<b>Description</b>	Sends a PPL Event Request Acknowledge to L5 Status given in Arg1
<b>Arg1</b>	Status
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Send BCC Channel Request to already active Cb
<b>Description</b>	Sends a Channel Request for already active Cb to BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Send BCC Channel Remove Forced
<b>Description</b>	Sends a Channel Remove to BCC not expecting ACK
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Clear L3 outgoing buffer
<b>Description</b>	This function will clean up the L3 outgoing buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Store Called Party IE Digits
<b>Description</b>	Stores digits in the called party IE from L4 in the digit array for this Cb
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>98</b>
<b>Name</b>	Store the Call Control Flag from L4
<b>Description</b>	Stores the call control flag from L4 for this Cb. This flag determines the type of Acknowledge to send, and where to send it (L4/L5)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Mark as Outgoing Call
<b>Description</b>	Set the Outgoing Call Flag for this call to TRUE
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Transfer Data from L3 incoming to L4 outgoing buffer
<b>Description</b>	This function will move/copy the L3 incoming buffer to the L4 outgoing buffer
<b>Arg1</b>	Move (0) or Copy (1)
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Transfer Data from L4 incoming to L3 outgoing buffer
<b>Description</b>	This function will move/copy the L4 incoming buffer to the L3 outgoing buffer
<b>Arg1</b>	Move (0) or Copy (1)
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Store L4 outgoing buffer
<b>Description</b>	This function stores the L4 outgoing buffer in temporary buffer to make outgoing buffer available
<b>Arg1</b>	Move (0) or Copy (1)
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>103</b>
<b>AF Name</b>	Restore the saved L4 outgoing buffer
<b>AF Description</b>	This function restores the L4 outgoing buffer that was stored in temporary buffer.
<b>Arg1</b>	Move (0) or Copy (1)
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>113</b>
<b>Name</b>	Load Pulse -Signal IE in L3 outgoing buffer
<b>Description</b>	Load the pulse-signal IE in the L3 outgoing buffer The Pulse Type is stored in the configuration byte given in Arg1 The number of pulses are given in Arg2
<b>Arg1</b>	Configuration byte storing the Pulse Type
<b>Arg2</b>	Number of Pulses
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>114</b>
<b>Name</b>	Load Steady Signal IE in L3 outgoing <steady signal type>
<b>Description</b>	This function will load the Steady signal IE with the specified steady signal type in the L3 outgoing buffer
<b>Arg1</b>	Pulse Type
<b>Arg2</b>	Number of Pulses
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>115</b>
<b>Name</b>	Load Pulsed Signal IE in L3 outgoing
<b>Description</b>	This function will load the Steady signal IE with the specified steady signal type in the L3 outgoing buffer
<b>Arg1</b>	Pulse Type
<b>Arg2</b>	Number of Pulses
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>116</b>
<b>Name</b>	Test if Pulse Type in Pulse-Signal IE in L3 incoming buffer matches value of configuration byte
<b>Description</b>	Test to see if the Pulse Type in Pulse-Signal IE in L3 incoming buffer matches value of configuration byte
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>119</b>
<b>Name</b>	Test Steady-Signal Type in Steady Signal IE in L4 incoming buffer
<b>Description</b>	Tests steady-signal type in the steady signal IE in L4 incoming message
<b>Arg1</b>	Steady Signal Type
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if the Steady-Signal Type matches, PPL Event INT Event 0 if the Steady-Signal Type does not match
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>120</b>
<b>Name</b>	Validate L4 message <msg event>
<b>Description</b>	Validates the message from L4 to L3P PSTN
<b>Arg1</b>	PPL Event Number that corresponds to the message from L4
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if the message is valid, PPL Event INT Event 0 if the message is invalid
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>121</b>
<b>Name</b>	Test if IE exists in L3 incoming message < IE type>
<b>Description</b>	This function checks whether the IE specified with IE type is present in L3 incoming message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if the IE exists, PPL Event INT Event 0 if the IE does not exist
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>122</b>
<b>Name</b>	Test if IE exists in L4 incoming message < IE type>
<b>Description</b>	This function checks whether the IE specified with IE type is present in L4 incoming message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if the IE exists, PPL Event INT Event 0 if the IE does not exist
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>123</b>
<b>Name</b>	Test Steady-Signal Type in Steady Signal IE in L3 incoming buffer
<b>Description</b>	This function tests the steady-signal type in the steady signal IE in L3 incoming message
<b>Arg1</b>	Steady Signal Type
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if the Steady-Signal Type matches, PPL Event INT Event 0 if the Steady-Signal Type does not match
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>124</b>
<b>Name</b>	Test if timeslot value is received from BCC
<b>Description</b>	This function tests the flag in V5 PSTN SP RW that indicates whether the Channel Request Acknowledge message has been received from BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 0 + value of Timeslot received from BCC in V5 PSTN SP RW
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>125</b>
<b>Name</b>	Test the Error code TLV from BCC
<b>Description</b>	This function tests the error code value in the Error Code TLV in the message from BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 0 + Value of error code
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>126</b>
<b>Name</b>	Test for termination character in L3 Digit Info
<b>Description</b>	This function tests if a termination character is present in the L3 Digit Info in the message from L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if termination character exists, PPL Event INT Event 0 if it does not exist
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>127</b>
<b>Name</b>	Test the Status Event in the Status Indication message from L3
<b>Description</b>	This function tests value of the status event in the Status Indication message from L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 0 + Value of status event
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>128</b>
<b>Name</b>	Test if waiting for deferred blocking
<b>Description</b>	This function tests the flag in V5 PSTN SP RW that indicates whether the user port should be blocked after the call on this user port is released.
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if Deferred Blocking in flag set
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>129</b>
<b>Name</b>	Test if Outgoing Call
<b>Description</b>	Outputs PPL Event INT Event 1 if Outgoing Call; Outputs PPL Event INT Event 0 if Incoming Call
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Outputs</b>	PPL Event INT Event 1 if Deferred Blocking in flag set
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>130</b>
<b>Name</b>	Load stored digit info into L4 outgoing buffer
<b>Description</b>	Loads the stored digit information into the L4 outgoing buffer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Set the deferred blocking enable flag
<b>Description</b>	Sets the deferred blocking enable flag for this PSTN Cb
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Update timeslot value from BCC
<b>Description</b>	This function updates the timeslot value in the PSTN CB from the value given in the BCC incoming message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>133</b>
<b>Name</b>	Set flag indicating timeslot is received from BCC
<b>Description</b>	This function sets the flag that indicates whether a valid timeslot value has been received from BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Store the Digit Info from L3 incoming buffer
<b>Description</b>	This function stores the digit info (digits etc.) that were received in the L3 incoming buffer
<b>Arg1</b>	<Event #>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Check if LE side
<b>Description</b>	Tests config for this V5 ID to determine if LE side
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Test the reject Cause in incoming message from BCC
<b>Description</b>	To test the reject cause sent in channel request reject and the channel indication reject messages from incoming from the BCC protocol; the function will return appropriate status code to the host
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

## L3P BCC (0x0092) Atomic Functions

---

**Purpose** This section describes the atomic functions for the L3P BCC PPL component.

**L3P BCC Atomic Functions** The following atomic functions are specific to the ISDN L3P BCC component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Send L3 ALLOCATION Request
<b>Description</b>	This function sends an Allocation Request to L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Send L3 ALLOCATION COMPLETE Request (AN only)
<b>Description</b>	This function sends an Allocation Complete Request to L3 ( <i>Allocation Confirm</i> message specifying Allocation Complete)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Send L3 ALLOCATION REJECT Request (AN only)
<b>Description</b>	This function sends an Allocation Reject Request to L3 ( <i>Allocation Confirm</i> message specifying Allocation Reject)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Send L3 DEALLOCATION Request
<b>Description</b>	This function sends a Deallocation Request message to L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Send L3 DEALLOCATION COMPLETE Request (AN only)
<b>Description</b>	This function sends a Deallocation Complete Request to L3 (Deallocation Confirm message specifying Deallocation Complete)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Send L3 AUDIT Request (AN only)
<b>Description</b>	This function sends a AUDIT Request to L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>57</b>
<b>Name</b>	Send Audit Complete to L3
<b>Description</b>	Sends Audit Complete message to Layer 3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Test for User Port match in Audit Complete message
<b>Description</b>	Tests for User Port match in the Audit Complete message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Send Clear Call Request to PSTN
<b>Description</b>	Sends a Clear Call Request to the PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Send PSTN Channel Request Ack
<b>Description</b>	This function sends a Channel Request Acknowledge message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send PSTN Channel Request Reject
<b>Description</b>	This function sends a Channel Request Reject message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>62</b>
<b>Name</b>	Send PSTN Channel Remove Ack
<b>Description</b>	This function sends a Channel Remove Acknowledge message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>63</b>
<b>Name</b>	Send PSTN Channel Remove Reject
<b>Description</b>	This function sends a Channel Remove Reject message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>64</b>
<b>Name</b>	Send PSTN Channel Indication Ack
<b>Description</b>	This function sends a Channel Indication Acknowledge message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>65</b>
<b>Name</b>	Send Layer Manager Deferred Link Blocked Acknowledge
<b>Description</b>	Sends a Deferred Link Blocked Acknowledge to Layer Manager
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>66</b>
<b>Name</b>	Send PSTN Channel Indication Reject
<b>Description</b>	This function sends a Channel Indication Reject message to the PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>68</b>
<b>Name</b>	Send PSTN Channel Allocated Indication
<b>Description</b>	This function sends a Channel Allocated Indication message to PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send PSTN Channel Allocation Reject
<b>Description</b>	Sends a Channel Allocation Reject to the PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send PSTN a BCC Error Indication <Cause>
<b>Description</b>	Sends PSTN a BCC Error Indication
<b>Arg1</b>	Cause: 1: Userport/Timeslot mismatch 2: Fault Message received from AN 3: Error Indication from Layer 3
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send L4 Router Port Request
<b>Description</b>	This function sends a Port Request message to L4 Router
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send L4 Router Reserve Port Request (AN only)
<b>Description</b>	This function sends a Reserve Port Request to L4 Router
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Check Status Indication for Reason
<b>Description</b>	Tests the Status Indication from L3 to determine if the reason specifies Audit Required
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Get BCC Reference Number
<b>Description</b>	Retrieves the BCC Reference Number
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Send PPL Event Indication with L4 Router reject status
<b>Description</b>	Tests the value of the reject status from L4 router and sends it to L5
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Send AN Fault Request to L3
<b>Description</b>	Sends a Fault Request to the peer entity; function is valid only on AN end
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Set the timeslot value from L4 incoming buffer
<b>Description</b>	This function sets the timeslot value equal to timeslot given in message from L4
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Set the timeslot value for this user port from V5.2 Timeslot/Userport table
<b>Description</b>	This function gets and sets the timeslot value from the user port ID by looking it up in the V5.2 Timeslots/Userport table
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Test the Error Code TLV in the message from PSTN
<b>Description</b>	This function tests the value of the Error Cause in the Error Code TLV in the incoming message from PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>106</b>
<b>Name</b>	Test if we are LE
<b>Description</b>	Tests to determine if this V5 ID is LE or not
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	If LE: PPL Event INT Event 1 If AN: PPL Event INT Event 0

<b>AF Number</b>	<b>107</b>
<b>Name</b>	Test if we are V5.2
<b>Description</b>	Tests to determine if this V5 ID is V5.2 or not
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	If LE: PPL Event INT Event 1 If AN: PPL Event INT Event 0

<b>AF Number</b>	<b>110</b>
<b>Name</b>	Test the Response Status from L4RTR
<b>Description</b>	Tests the value of the Response Status TLV received from L4RTR
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>111</b>
<b>Name</b>	Test the More Response Status from L4RTR
<b>Description</b>	Tests the value of the More Response Status TLV received from L4RTR
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

## L3P MGR (0x0093) Atomic Functions

---

**Purpose** This section describes the atomic functions for the L3P MGR PPL component.

**L3P MGR Atomic Functions** The following atomic functions are specific to the ISDN MGR PPL component.

<b>AF Number</b>	<b>51</b>
<b>Name</b>	Query All Span Status
<b>Description</b>	This function queries All Span Status
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>52</b>
<b>Name</b>	Check whether all spans are alive
<b>Description</b>	This function checks whether all spans are alive
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	PPL Int Event 1 = Success; PPL Int Event 0 = Failure

<b>AF Number</b>	<b>53</b>
<b>Name</b>	Update the Layer 3P L Manager Link Status
<b>Description</b>	This function updates the Layer 3P L Manager Link Status
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>54</b>
<b>Name</b>	Configure general for Envelope Function Layer
<b>Description</b>	This function configures general for Envelope Function Layer
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Configure general for LAPV
<b>Description</b>	This function configures general for LAPV
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>56</b>
<b>Name</b>	Configure general for L3
<b>Description</b>	This function configures general for L3
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>57</b>
<b>Name</b>	ConfigureV5 Interface
<b>Description</b>	This function configuresV5 Interface
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>58</b>
<b>Name</b>	Configure User Ports
<b>Description</b>	This function configures User Ports
<b>Arg1</b>	0 = Configure User Ports; 1 = Add User Ports
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>59</b>
<b>Name</b>	Set Don't do General Configuration Global Data Variable
<b>Description</b>	Function sets Don't do General Configuration Global Data Variable
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Clear Don't do General Configuration Global Data Variable
<b>Description</b>	Function clears Don't do General Configuration Global Data Variable
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Check Don't do General Configuration Global Data Variable
<b>Description</b>	Function checks Don't do General Configuration Global Data Variable
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	PPL Int Event 1 = Set; PPL Int Event 0 = Not Set

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send Startup Request to System Management
<b>Description</b>	This function sends Startup Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send Startup Event PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Startup Event PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: Startup Layer-2 Activation Fail 2: Startup Variant ID Mismatch 3: Startup Link ID Failed 4: Startup Complete 5: Startup Failed
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send Delete V5 Interface message to System Management
<b>Description</b>	This function sends Delete V5 Interface message to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send V5 Interface Restart Request to System Management
<b>Description</b>	Sends V5 Interface Restart Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send Restart Event PPL Indication to host <Arg 1>
<b>Description</b>	Sends V5 Interface Restart Request to System Management
<b>Arg1</b>	1 = Restart timer expired, Restart failed; 2 = Restart complete
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send V5 Interface Status to host
<b>Description</b>	Sends V5 Interface Status to host
<b>Arg1</b>	0 = Interface is not up; 1 = Interface ID up
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>77</b>
<b>Name</b>	Set the interface State
<b>Description</b>	Sets the interface State
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>78</b>
<b>Name</b>	Print Interface Alive message
<b>Description</b>	Prints Interface Alive message
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>81</b>
<b>Name</b>	Send Link Not in Operation message to System Management
<b>Description</b>	Sends Link Not in Operation message to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>82</b>
<b>Name</b>	Send Blocked Link Indication to System Management
<b>Description</b>	Sends Blocked Link Indication to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>83</b>
<b>Name</b>	Send Deferred Link Block Request to System Management
<b>Description</b>	Sends Deferred Link Block Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>84</b>
<b>Name</b>	Send Non-Deferred Link Block Request to System Management
<b>Description</b>	Sends Non-Deferred Link Block Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Send Unblock Link Request to System Management
<b>Description</b>	Sends Unblock Link Request to System Management
<b>Arg1</b>	0 = Use the current Link 1 = Use other Link (Useful only with Primary and Secondary links, such as when Primary Link is blocked and Secondary C-Channel is active. For switchover when the current link is Secondary, this option sends a Link Unblock message to the Primary link.
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>86</b>
<b>Name</b>	Send Link in Operation message to System Management
<b>Description</b>	Sends Link in Operation message to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>87</b>
<b>Name</b>	Send Add Link Request to System Management
<b>Description</b>	Sends Add Link Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>88</b>
<b>Name</b>	Send Delete Link Request to System Management
<b>Description</b>	Sends Delete Link Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>89</b>
<b>Name</b>	Send Block Indication to System Management from timer expiry
<b>Description</b>	Sends Block Indication to System Management from timer expiry
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Send Link MDU PPL Event Indication to host <Arg1>
<b>Description</b>	Sends Link MDU PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1Link operational 2 Link not operational 3Link ID Required 4 Link ID Request 5 Link ID Release Indication 6 Link ID Reject Indication 7Link ID Failure 8 Link Block Indication 9Link Unblock Indication 10Link Unblock Request 11Link Deferred Block Request 12 Link No-deferred Block Request
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Send Data Link Release Event PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Data Link Release Event PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1:Data link release received, autonomous switchover initiated 2:Datalink down Indication, standby not configured 3:Data link release received, autonomous switchover to be initiated on getting control protocol DL release 4:BCC DL release, audit connection requested.
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Send Link Status to host
<b>Description</b>	Sends Link Status to host
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>93</b>
<b>Name</b>	Send Layer 3P PSTN Link Blocked Request
<b>Description</b>	Sends Layer 3P PSTN Link Blocked Request
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>94</b>
<b>Name</b>	Send Link Down message to Layer 3P PSTN
<b>Description</b>	Sends Link Down message to Layer 3P PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>95</b>
<b>Name</b>	Send Deferred Link Block Request to Layer 3P PSTN
<b>Description</b>	Sends Deferred Link Block Request to Layer 3P PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>96</b>
<b>Name</b>	Send Non-Deferred Link Block Request to Layer 3P PSTN
<b>Description</b>	Sends Non-Deferred Link Block Request to Layer 3P PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>97</b>
<b>Name</b>	Send User Port message to Layer 3P PSTN
<b>Description</b>	Sends User Port message to Layer 3P PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>99</b>
<b>Name</b>	Send Channel Status to Layer 4 for all timeslots
<b>Description</b>	Sends Channel Status to Layer 4 for all timeslots
<b>Arg1</b>	0 = Channel Out Of Service 1 = Channel In Service 2 = Channel Blocked 3 = Clear Channel
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>100</b>
<b>Name</b>	Set Link Status
<b>Description</b>	Sets Link Status
<b>Arg1</b>	Value
<b>Arg2</b>	Bit Mask
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>101</b>
<b>Name</b>	Return C Channel Status to Link
<b>Description</b>	Returns C Channel Status to Links
<b>Arg1</b>	Value
<b>Arg2</b>	Bit Mask
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>102</b>
<b>Name</b>	Check Primary or Secondary Link Up
<b>Description</b>	Checks Primary or Secondary Link Up
<b>Arg1</b>	Value
<b>Arg2</b>	Bit Mask
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>103</b>
<b>Name</b>	Test Link Status
<b>Description</b>	Tests Link Status
<b>Arg1</b>	Bit Mask
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>104</b>
<b>Name</b>	Send all configured links status to System Management
<b>Description</b>	Sends all configured links status to System Management
<b>Arg1</b>	0 = Block only locally-blocked Links 1 = Unblock Unblocked Links
<b>Arg2</b>	Bit Mask
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>105</b>
<b>Name</b>	Send MPH AIs to System Management
<b>Description</b>	Sends MPH AIs to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>111</b>
<b>Name</b>	Check whether other C Channel is active
<b>Description</b>	Checks whether other C Channel is active (other than the channel that is currently active)
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>112</b>
<b>Name</b>	Send C Channel Status to host
<b>Description</b>	Sends C Channel Status to host
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>113</b>
<b>Name</b>	Send Accelerated Port Alignment Request to System Management
<b>Description</b>	Sends Accelerated Port Alignment Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>114</b>
<b>Name</b>	Send User Port Block Request to System Management
<b>Description</b>	Sends User Port Block Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>115</b>
<b>Name</b>	Send User Port Unblock Request to System Management
<b>Description</b>	Sends User Port Unblock Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>116</b>
<b>Name</b>	Send User Port Block Indication to System Management
<b>Description</b>	Sends User Port Block Indication to System Management
<b>Arg1</b>	0 = Send Block Indication 1 = Send Don't Locally block Indication
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>117</b>
<b>Name</b>	Send Delete User Port message to System Management
<b>Description</b>	Sends Delete User Port message to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>118</b>
<b>Name</b>	Send User Port Status Request to System Management
<b>Description</b>	Sends User Port Status Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>119</b>
<b>Name</b>	Send MPH PPL Event Indication to host <Arg1>
<b>Description</b>	Sends MPH PPL Event Indication to host
<b>Arg1</b>	<p>Any one of the following:</p> <ul style="list-style-type: none"> <li>1 Port Unblock Request</li> <li>2 Port Block Indication</li> <li>3 Port Unblock Indication</li> <li>4 Port Block Request</li> <li>5 ISDN-PRI AN Maintenance</li> <li>6 ISDN-PRI AN Maintenance</li> <li>7 ISDN-PRI AN Maintenance</li> <li>8 ISDN-PRI AN Maintenance</li> <li>9 ISDN-PRI AN Maintenance</li> <li>10 ISDN-PRI AN Maintenance</li> <li>11 ISDN-PRI AN Maintenance</li> <li>12ISDN-block D-channel from user</li> <li>13 ISDN Unblock D-channel</li> <li>14 User Failure Indication from Userport</li> <li>15 Network failure indication</li> <li>16Access Activated</li> <li>17 Access Deactivated</li> <li>18Grading Information</li> <li>19Reception of FE2</li> <li>20DS Active</li> <li>21Indication of DL Failure</li> <li>22Indication of LOS/LFA</li> <li>23Reception of FE101</li> <li>24Reception of FE105</li> <li>25Unsuccessful Activation Attempt</li> <li>26Access Activation by User</li> </ul>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>120</b>
<b>Name</b>	Send Userport Status to host
<b>Description</b>	Sends Userport Status to host
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>121</b>
<b>Name</b>	Send Port Error PPL Event Indication to host <Arg 1>
<b>Description</b>	Sends Port Error PPL Event Indication to host
<b>Arg1</b>	1 = Port FSM received an invalid Event
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>122</b>
<b>Name</b>	Test whether call is active on the Userport
<b>Description</b>	Tests whether call is active on the Userport or not
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>123</b>
<b>Name</b>	Send Block Indications to System Management for the ports that are locally blocked
<b>Description</b>	Sends Block Indications to System Management for the ports that are locally blocked
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>124</b>
<b>Name</b>	Send Unblock Request to System Management for all ports, then send Block Indications to ports that are locally blocked
<b>Description</b>	Sends Unblock Request to System Management for all ports, then send Block Indications to ports that are locally blocked
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>131</b>
<b>Name</b>	Send Switchover Request to System Management
<b>Description</b>	Sends Switchover Request to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>132</b>
<b>Name</b>	Send Switchover Complete message to System Management
<b>Description</b>	Sends Switchover Complete message to System Management
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>133</b>
<b>Name</b>	Send Switchover Event PPL Indication to host <Arg 1>
<b>Description</b>	Sends Switchover Event PPL Indication to host
<b>Arg1</b>	Any of these: 1 = Layer Manager to perform Envelope function layer SAP config. 2 = Switchover reject received 3 = Switchover completed 4 = Switchover failed
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>134</b>
<b>Name</b>	Send Reconfigure to Ev Layer for Switchover
<b>Description</b>	Sends Reconfigure to Ev Layer for Switchover
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>135</b>
<b>Name</b>	Straighten out Ev SAPs
<b>Description</b>	Straightens out Ev SAPs
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>136</b>
<b>Name</b>	Test whether Ev crossed
<b>Description</b>	Tests whether Ev crossed
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>137</b>
<b>Name</b>	Toggle Ev SAPs
<b>Description</b>	Toggles Ev SAPs
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>138</b>
<b>Name</b>	Check for reasons for switchover fail
<b>Description</b>	Checks for reasons for switchover fail. Returns 0 (indicating failure) in any of these cases: Both C-Channels are Dead Both C-Channels are blocked One C-Channel is Dead and the other is blocked
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>141</b>
<b>Name</b>	Send Statistics Error PPL Event Indication to host <Arg 1>
<b>Description</b>	Sends Statistics Error PPL Event Indication to host <Arg 1>
<b>Arg1</b>	1 = Invalid Entity, Instance 2 = Invalid Port ID
<b>Arg2</b>	<Not Used>
<b>Returns</b>	PPL Int Event 1 (Success); PPL Int Event 0 (Failure)

<b>AF Number</b>	<b>142</b>
<b>Name</b>	Send Statistics request to lower layers
<b>Description</b>	Sends Statistics request to lower layers (System Management, Envelope layer, or LAPV)
<b>Arg1</b>	1 = Invalid Entity, Instance 2 = Invalid Port ID
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>143</b>
<b>Name</b>	Send Statistics data to host
<b>Description</b>	Sends Statistics data to host
<b>Arg1</b>	0 = Vf Layer Statistics 1 = Ev layer Statistics 2 = LAPV Statistics
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>146</b>
<b>Name</b>	Send Status Indication to Layer 3P BCC
<b>Description</b>	Sends Status Indication to Layer 3P BCC
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>147</b>
<b>Name</b>	Send Status Error PPL Event Indication to host <arg1>
<b>Description</b>	This function sends Status Error PPL Event Indication to host
<b>Arg1</b>	Any of the following: 1Invalid entity and Instance 2Invalid Interface Number 3Invalid port type 4Invalid port ID 5Invalid Link number
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>149</b>
<b>Name</b>	Process the LAPV Status Indication

<b>Description</b>	Processes the LAPV Status Indication
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>151</b>
<b>Name</b>	Set HDLC Driver Action
<b>Description</b>	Sets HDLC Driver Action
<b>Arg1</b>	0 = Disable 1 = Enable 2 = Disable All
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>152</b>
<b>Name</b>	Disable HDLC Driver for blocked or dead C Channels
<b>Description</b>	Disables HDLC Driver for blocked or dead C Channels
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>153</b>
<b>Name</b>	V5 MGR test whether V5 Subscriber ID exists
<b>Description</b>	Tests whether V5 Subscriber ID exists
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>154</b>
<b>Name</b>	V5 MGR send PPL Event Indication to Layer 5 with V5 Subscriber ID
<b>Description</b>	Sends PPL Event Indication to Layer 5 with V5 Subscriber ID
<b>Arg1</b>	Ev Indication
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>155</b>
<b>Name</b>	V5 MGR send PPL Event Indication to Layer 5 with no ICB
<b>Description</b>	V5 MGR sends PPL Event Indication to Layer 5 with no ICB
<b>Arg1</b>	Event Indication Number
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>161</b>
<b>Name</b>	Send Management Port PPL Event Indication to host <Arg 1>
<b>Description</b>	Sends Management Port PPL Event Indication to host
<b>Arg1</b>	1 = Event out of sequence 2 = Saved Event already reached limit 3 = Timer T2 expired twice
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>162</b>
<b>Name</b>	Send Management Common PPL Event Indication to host <Arg 1>
<b>Description</b>	Sends Management Common PPL Event Indication to host
<b>Arg1</b>	1 = Event out of sequence 2 = Saved Event already reached limit 3 = Timer T2 expired twice
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>163</b>
<b>Name</b>	Send Event Message Decode PPL Indication to host <Arg 1>
<b>Description</b>	Sends Event Message Decode PPL Event Indication to host
<b>Arg1</b>	1 = Message Decoding Error 2 = Invalid message 3 = Invalid message length 4 = Protocol discriminator error 5 = Layer 3 address error 6 = Invalid message type 7 = Invalid message content 8 = Invalid optional IE
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>164</b>
<b>Name</b>	Send Control Request OK PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Control Request OK PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: All Ports Blocked 2: All Linked Blocked 3: Port/All Port Deleted 4: Interface Deleted 5: Accelerated Port Alignment OK 6: All Links/Link Deleted
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>165</b>
<b>Name</b>	Send Config OK PPL Event Indication to host
<b>Description</b>	This function sends Config OK PPL Event Indication to host
<b>Arg1</b>	1: V5 General CFG OK 2: V5 TSAP CFG OK 3: V5 Interface CFG OK 4: V5 Link CFG OK 5: V5 Userport CFG OK
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>166</b>
<b>Name</b>	Send Acknowledgment to host in response to PPL Event Request
<b>Description</b>	Sends Acknowledgment to host in response to PPL Event Request
<b>Arg1</b>	0 = Positive Acknowledgment 1 = Invalid Event
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>168</b>
<b>Name</b>	Send Layer 4 Primitive Failed PPL Event Indication to host
<b>Description</b>	Sends Layer 4 Primitive Failed PPL Event Indication to host
<b>Arg1</b>	0 = Reasons Available 1 = Reasons Unavailable
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>169</b>
<b>Name</b>	Send Status Indication to Layer 3P PSTN
<b>Description</b>	Sends Status Indication to Layer 3P PSTN
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>170</b>
<b>Name</b>	Send Invalid State PPL Event Indication to host <Arg1>
<b>Description</b>	Sends Invalid State PPL Event Indication to host
<b>Arg1</b>	1 = BCC/Protection protocol State event matrix received invalid event
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>171</b>
<b>Name</b>	Send UI Event PPL Event Indication to host <Arg1>
<b>Description</b>	Sends UI Event PPL Event Indication to host
<b>Arg1</b>	1 = Layer 4 primitive failed, requested TSAP not found
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>172</b>
<b>Name</b>	Send LVT INT Down PPL Event Indication to host <Arg1>
<b>Description</b>	Sends LVT INT Down PPL Event Indication to host
<b>Arg1</b>	1 = Control Protocol TCON Timer expired
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>173</b>
<b>Name</b>	Send Common Event PPL Event Indication to host <Arg1>
<b>Description</b>	Sends Common Event PPL Event Indication to host
<b>Arg1</b>	1 = Ready for Reprovisioning 2 = Not ready for reprovisioning 3 = Variant and ID received 4 = Switchover to new variant 5 = Cannot reprovision 6 = Verify reprovisioning 7 = reprovisioning started
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>174</b>
<b>Name</b>	Send message Send PPL Event Indication to host <Arg1>
<b>Description</b>	Sends message Send PPL Event Indication to host
<b>Arg1</b>	1 = Message send failure, Invalid protocol FSM State 2 = Protection protocol message send failure, both DLs down
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>175</b>
<b>Name</b>	Send message Received PPL Event Indication to host <Arg1>b
<b>Description</b>	Sends message Received PPL Event Indication to host
<b>Arg1</b>	1 = Message received an Invalid protocol FSM State
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>176</b>
<b>Name</b>	Send timer expired PPL Event Indication to host <Arg1>
<b>Description</b>	Sends timer expired PPL Event Indication to host
<b>Arg1</b>	1= Secondary expiry of Link control protocol timer
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>177</b>
<b>Name</b>	Send L1 FSM PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends L1 FSM PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: Layer Manager should issue MPH ID to Layer 1 2: Layer Manager should issue MPH IDR to Layer 1 3: Layer Manager should issue MPH NOR to Layer 1
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>178</b>
<b>Name</b>	Send DSFE PPL Event Indication to host <Arg1>
<b>Description</b>	Sends DSFE PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: ISDN activate access 2: ISDN deactivate access
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>179</b>
<b>Name</b>	Send Timer PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Timer PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: PSTN Timer T3 expiry 2: PSTN Timer T4 expiry
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>180</b>
<b>Name</b>	Send Protection Layer PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Protection Layer PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1: Reset SN error 2: General protocol error message 3: Notification to layer management of SN misalignment. 4: Reset SN Indication after action on reset SN Request 5: Reset SN acknowledge reception Indication
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>181</b>
<b>Name</b>	Send Protocol Error Message PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Protocol Error Message PPL Event Indication to host
<b>Arg1</b>	1 = BCC Protocol Error message received
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>182</b>
<b>Name</b>	Send Configuration Error PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Configuration Error PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1 Invalid Entity, Instance 2 Invalid Element Type 3 Configuration already done 4 Cannot Allocate Memory 5 Invalid Interface Number 6 Link Already Present 7 Maximum links already present 8 Invalid Port Type 9 Ports exceed maximum ports 10 Port already present 11 Port insert into hash list failed
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>183</b>
<b>Name</b>	Send Control Error PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Control Error PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1Invalid Entity, Instance 2Invalid Interface number 3Invalid Element, Type 4Invalid unsolicited action 5Invalid trace generation action 6Invalid sub-action parameter 7Block port error 8Delete port error 9Delete interface error 10Delete link error 11Invalid port ID 12Block Link error
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>184</b>
<b>Name</b>	Send Bind error PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Bind error PPL Event Indication to host
<b>Arg1</b>	1 = Layer 4 unable to bind/unbind to Layer 3
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>185</b>
<b>Name</b>	Send Common Error PPL Event Indication to host <Arg1>
<b>Description</b>	This function sends Common Error PPL Event Indication to host
<b>Arg1</b>	Any one of the following: 1:V5.1FSM getting a bad input event 2:Variant and ID-mismatch 3:V5.1FSM reporting control protocol DL failure 4:V5.1 FSM timer expiry 5:V5.2 interface FSM receiving bad Input event. 6:Accelerated port Alignment failed.
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>191</b>
<b>Name</b>	Check whether LE Side
<b>Description</b>	Checks whether LE side
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

<b>AF Number</b>	<b>201</b>
<b>Name</b>	Sec C Channel to change the State to Alive from Active
<b>Description</b>	Allows the Sec C Channel to change state to Alive from Active only when the primary C Channel is already active, and also to set both the C Channel states appropriately
<b>Arg1</b>	<Not Used>
<b>Arg2</b>	<Not Used>
<b>Returns</b>	Success or Failure

# 8 VDAC/IP Network Interface Atomic Functions

**Purpose** This chapter describes the atomic functions that can be used with VoIP modules on a Voice Data Access Concentrator (VDAC)-ONE card or IP Network Interface Series 2 card. The atomic functions in this chapter are used with the component ID, 0x009C.

**Atomic Functions**

<b>AF Number</b>	<b>46</b>
<b>Name</b>	Enable multipurpose timer
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Enable 1 of 3 multipurpose timers indicated in Arg 1. The value of the timer is stored in the timer indicated in Arg 2.
<b>Arg 1</b>	Timer Number (1 – 3)
<b>Arg 2</b>	Timer ID (1 – 100)

<b>AF Number</b>	<b>47</b>
<b>Name</b>	Disable multipurpose timer
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Disable 1 of 3 multipurpose timers indicated in Arg 1.
<b>Arg 1</b>	Timer number (1 – 3)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>55</b>
<b>Name</b>	Initiate a channel purge
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Initiate a channel purge and pass the purge reason indicated in Arg 1 to Layer 4.
<b>Arg 1</b>	Purge reason (0 – 255)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>60</b>
<b>Name</b>	Send DSP a <i>Request Resource</i> message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Send a request to DSP to connect to an available DSP channel.
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>61</b>
<b>Name</b>	Send DSP a <i>Release Resource</i> message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Send a request to DSP to disconnect from a previously-connected DSP channel
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>70</b>
<b>Name</b>	Send Layer 4 a <i>Connect</i> message (Answer)
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a <i>Connect</i> message to Layer 4.
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>71</b>
<b>Name</b>	Send Layer 4 an <i>Access Denied</i> message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Send an <i>Access Denied</i> message to Layer 4 due to a failed connection attempt. Failure reason is indicated in Arg 1.
<b>Arg 1</b>	Status (0 – 255)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>72</b>
<b>Name</b>	Send Layer 4 an alerting message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends an alerting message to Layer 4
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>73</b>
<b>Name</b>	Send Layer 4 a clear Ack.message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a clear request Ack. to Layer 4, with successful and lost RTP packets
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>74</b>
<b>Name</b>	Send Layer 4 an <i>In Service</i> message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends an <i>In Service</i> message to Layer 4
<b>Arg 1</b>	Reserved

<b>Arg 2</b>	Reserved
--------------	----------

<b>AF Number</b>	<b>75</b>
<b>Name</b>	Send Layer 4 a <i>Disconnect</i> message
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a <i>Disconnect</i> message to Layer 4
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>76</b>
<b>Name</b>	Send Layer 4 a <i>Clear Request</i> acknowledgment without data
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a <i>Clear Request</i> acknowledgment to Layer 4, with no RTP packets status
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>80</b>
<b>Name</b>	Send outsize control response
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a response to the <i>Outsize Control</i> message.
<b>Arg 1</b>	Status (0 – 255)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>85</b>
<b>Name</b>	Store message into a temporary storage buffer
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Saves the current message, including attached data, into a temporary storage buffer to use later.

<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>90</b>
<b>Name</b>	Send Layer 5 a PPL Event Request Acknowledgment
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a PPL Event Request acknowledgment to Layer 5.
<b>Arg 1</b>	Status (0 – 255)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>91</b>
<b>Name</b>	Send Layer 5 a PPL Event Indication
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a PPL Event Indication to Layer 5.
<b>Arg 1</b>	Event Number (0 – 255)
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>92</b>
<b>Name</b>	Send Layer 5 a PPL Event Indication with General Purpose Register value
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sends a PPL Event Indication to Layer 5 along with the value of the specified General Purpose Register.
<b>Arg 1</b>	Event Number (0 – 255)
<b>Arg 2</b>	General Purpose Register number (0 – 53)

<b>AF Number</b>	<b>95</b>
<b>Name</b>	DSPM setup resource
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Sets up the resources required to communicate with the DSP Manager. This function should be called after the DSP Manager acknowledges the DSP resource request with a response of “DSP Resource Available”
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>96</b>
<b>Name</b>	DSPM connect resource to TDM
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Connects a DSP resource to the TDM bus. This function should be called after a resource has been setup.
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>97</b>
<b>Name</b>	DSPM disconnect resource from TDM
<b>Type</b>	Normal/Blocking/Test: Normal

<b>Description</b>	Disconnects a DSP resource to the TDM bus.
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved

<b>AF Number</b>	<b>98</b>
<b>Name</b>	DSPM remove resource.
<b>Type</b>	Normal/Blocking/Test: Normal
<b>Description</b>	Removes a DSP resource. This function should be called after all communication with the DSP Manager is completed.
<b>Arg 1</b>	Reserved
<b>Arg 2</b>	Reserved