ERICSSON

AXC 470 IDN
Software Functional Specification

Author: Kevin Stocksdale

Intended Audience: **Ericsson, Jade, and Reuters use only**

Approvals:

Software Engineer	Date
Program Manager	Date
Reuters Approval	Date

Document number: 1802305(C)
Template Number: 1100144(D)

TABLE OF CONTENTS

0 Revision History

A
11/30/2000
Kevin Stocksdale
Initial release.

B
12/12/2000
Kevin Stocksdale
Added the official Ericsson document number (1802305) to the title page and all the page headers.
Updated section 1.2 Scope to remove the statement indicating total bandwidth of both DUWAN ports is limited to 8.192 Mbps
Updated section 2.1 Related Project Documentation to reference the latest release (December V1.2) of the Software Requirements Specification.

C
12/20/2000
Kevin Stocksdale
Updated section 4.1.2.8 Automatic T1 Timer Calculation to reference the **set monitor lapb t1 variance** operator command
Updated section 7.1.17 Set Monitor LAPB Utilization Speed to reflect the maximum speed.
Added section 7.1.19 Set Monitor LAPB T1 Variance to describe the omitted command.
Added LAPB T1 Variance parameter to sample output display in section 7.1.26 Display monitor parameter table.
Updated section 7.2.1 MIB Organization to reflect the MIB variable for the LAPB T1 Variance parameter.
Updated section A.1 CLISPEC/CLIGEN Input to reference the client object for the LAPB T1 Variance parameter.

1 Introduction

1.1 Purpose

This document serves as the functional specification for the enhancements to Ericsson's AXC 470 for application in the Reuters IDN (Integrated Data Network) environment. The goal of this document is to provide detailed background of the existing Reuters specific features (software and hardware) that are being merged from the Amazon and Amazon Rapids to the AXC 470 platform.

1.2 Scope

The existing design of the Reuters specific features was initiated and implemented several years ago, using the 11.1 software baseline and executes on the Amazon and Amazon Rapid hardware platforms. Due to a projected increase of traffic on Reuters IDN networks, the need for an increased processing and buffering capability exists. The AXC 470 platform using the 12.0 software baseline was selected due to its three-fold increase in packet handling capabilities when compared to the Amazon product. The AXC 470 features the following:

- One IDT 64475 RISC processor running at 266 MHz.
- One DIMM socket for memory: 16 MB, expandable to 128MB.
- One compact flash module: 8 MB.
- One console port supporting RS-232. The DCE port on the back panel provides direct connection to a serial PC port or terminal.
- Two 10baseT/100baseTX Ethernet ports.
- Two, four, or six ports with dual E1/T1 and/or dual Universal WAN interfaces. The maximum speed per UWAN port is 8.192 Mbps.
- One Real Time Clock providing the date and time of day. Time of day is maintained for 72 hours in the event of a power outage.

Reuters has funded an effort to apply the "Reuters specific features" to the AXC 470 hardware and associated 12.0 software baseline. The resulting new product is known as the AXC 470 IDN, as the intended use of this product is in Reuters Integrated Data Network (IDN).

1.3 Marketing Requirements Summary

This effort is not driven by marketing, but by the customer (Reuters).

1.4 Feature Summary

The following summary of features was derived as a direct response to Reuters Software Requirements Specification (SRS):

- The Amazon and Amazon Rapid hardware enhancements that allow the software to output 5hz signals on two pins on the Console Monitor/Maintenance connector will be accommodated on the standard AXC 470 hardware.

- All of the Reuters specific enhancements (features and bug fixes) which comprise the 11.1.13.7 software release will be applied to the standard 12.0 software baseline. This includes, but is not limited to, the following Reuters specific features:

  - Live/Standby and Alarm Poll (see section 4.1.2.1)
  - Additional TRAP Messages (see section 4.1.2.2)
  - Enhance FDB Range Specification (see section 4.1.2.3)
  - Multi-Page Continuous Status Monitor (see section 4.1.2.4)
  - HTML Status/Error Monitor Pages (see section 4.1.2.5)
  - Hello Polling (see section 4.1.2.6)
  - Packet Prioritization (see section 4.1.2.7)
  - Automatic T1 Timer Calculations (see section 4.1.2.8)
  - Automatic Configuration (see section 4.1.2.9)
  - Multilink Frame Packing (see section 4.1.2.10)
  - Multilink Frame Recovery (see section 4.1.2.11)
  - Additional User Commands (see section 7.1.1 through 7.1.39)

- The following "standard" features of the 12.0 software baseline are of specific interest to Reuters and their function must not be compromised in the Reuters Enhanced 12.0 software baseline:

  - BOOTP Client
  - Flash File System
  - ACSII Script File Utility
  - TELNET Server Capability
  - TFTP Capability
  - Advanced Memory Management
  - HTTP/HTML Server Capability
  - Alarm Management Utility

- To ease the burden of field upgrades, the software registration key functionality will be removed from the 12.0 software baseline.

- The buffer allocation will be optimized for maximize LAN burst handling.

1.5 Performance Requirements

The performance of the AXC 470 hardware and Enhanced 12.0 software should be no less than the Amazon Rapids hardware and Enhanced 11.1 software.

1.6 Certifications Required

The certifications of the AXC 470 hardware and Enhanced 12.0 software should be no less than the Amazon Rapids hardware and Enhanced 11.1 software. Additionally, the hardware and software must successfully pass the tests described in the "ACC Amazon Enhancements Acceptance Test Specification" document developed by Reuters Ltd.

2 Reference Documents

2.1 Related Project Documents

Hoy, P., "RTTG LAN PROJECTS GROUP Software Requirements Specification ERICSSON AMAZON 470 LAN Bridge", V1.2, December 2000

2.2 Existing Documentation

Acorn, T., "ACC Amazon Enhancements Acceptance Test Specification", Version 1.0, September 1998

Stocksdale, K., "Tavarua & Tavarua Rapid Software Functional Specification", Ericsson Document Number 1800775(F), 1999

Roselinsky, M., Ericsson Functional Specification Template, Ericsson Document Number 1100144(A), 1993

2.3 Technical Reference

None.

2.4 Document Review Meeting Minutes

None.

3 Product Description

The primary purpose for this new product is to forward and filter (bridge) Ethernet frames from a local regional data center to a remote regional data center in an efficient and error free environment with minimal disruption of service in the event of any physical failures. The AXC 470 will be configured with two LAN interfaces and a minimum of two WAN interfaces. One LAN serves the purpose of the forwarding Market Stream Data and filtering everything else, the other LAN interface serves the purpose of local network management. All WAN interfaces are configured as a single Multilink group for load sharing and redundancy. Two AXC 470s will be installed at each regional data center. The second AXC 470 at each site provides a "Hot Spare" capability which is automatically switched into service in the event of a failure with the Primary bridge.

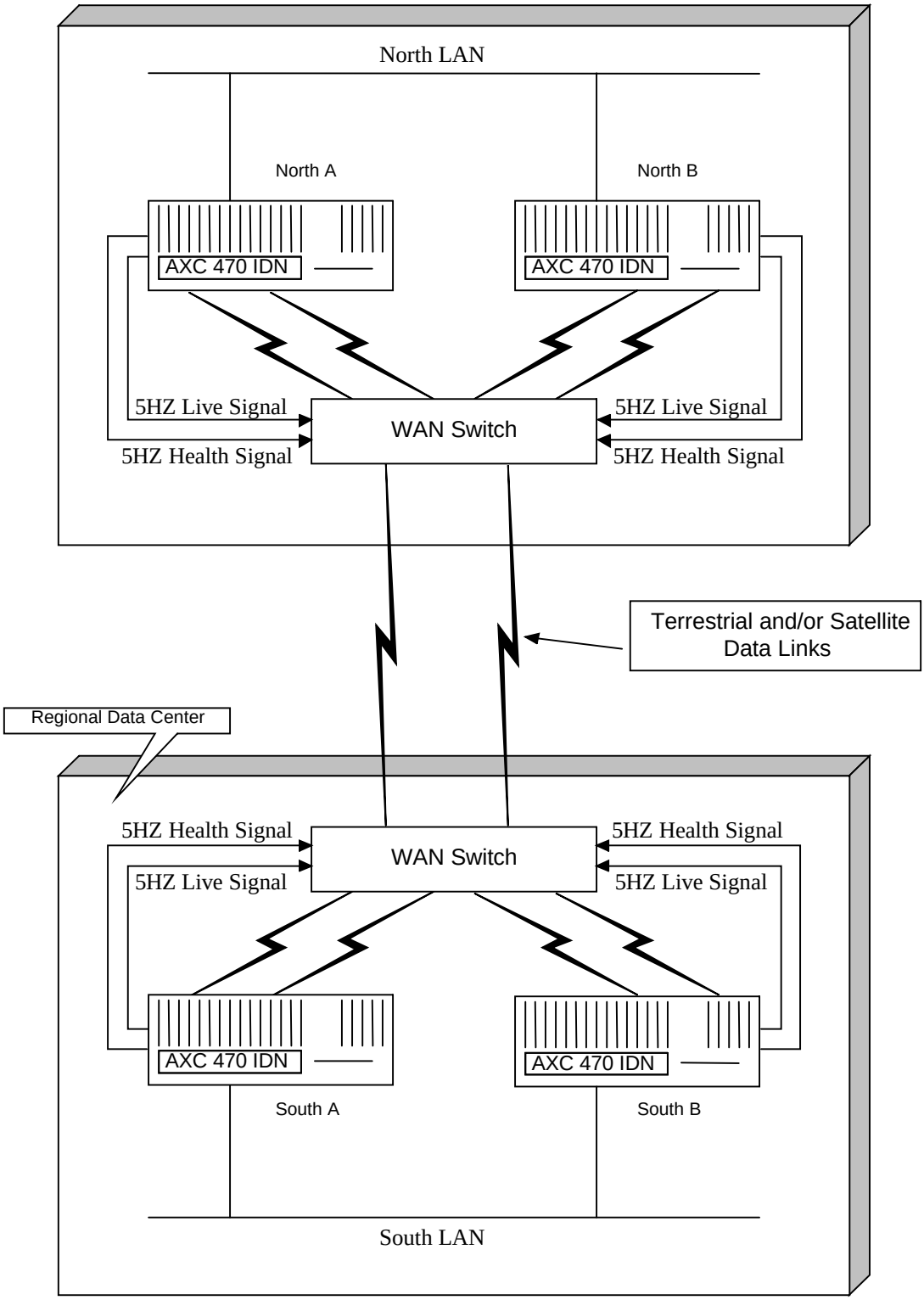
Modifications will be made to the standard AXC 470 hardware to allow the software executing on the AXC 470 to output 5hz signals on two pins on the Console Monitor/Maintenance connector. Modifications will also be made to the standard AXC 470 software to output the 5hz signals under normal operating condition, and to cease the 5hz signals when abnormal operating conditions occur.

The 5hz signals are wired (using a special cable designed by Reuters) to an external WAN Switch (also designed by Reuters) as depicted in diagram on the following page. The primary purpose of the WAN Switch is to physically switch the connection to the remote site (via the Terrestrial and/or Satellite Data links) to either the Primary or Secondary AXC 470 (via the Dual WAN interface connectors) at the local site.

One of the signals, known as the Live Signal, causes the external WAN Switch to switch when the signal ceases. This signal is regarded as a "heartbeat" of the AXC 470. Refer to the diagram on the following page in the following example:

> The AXC 470 labeled "North A" has established connectivity with the remote partner (lets say the AXC 470 labeled "South A"). The "North A" Bridge is powered off and the loss of the Live Signal from the "North A" Bridge will cause the WAN Switch at the North Site to switch. This allows connectivity between the Bridge labeled "North B" and the remote partner (the Bridge labeled "South A").

The other signal, known as the Health Signal, causes the external WAN Switch to display an audible and/or visual alarm when the signal ceases.


North LAN
North A
North B
AXC 470 IDN
AXC 470 IDN
5HZ Live Signal
5HZ Health Signal
WAN Switch
5HZ Live Signal
5HZ Health Signal
Terrestrial and/or Satellite Data Links
Regional Data Center
5HZ Health Signal
5HZ Live Signal
WAN Switch
5HZ Health Signal
5HZ Live Signal
AXC 470 IDN
AXC 470 IDN
South A
South B
South LAN

4 Functional Overview

4.1 Current Support

This section documents the currently supported features (hardware and software) of the Amazon hardware and Enhanced 11.1 software which are being ported to the AXC 470 hardware with 12.0 software.

4.1.1 Hardware Enhancements

4.1.1.1 IDN Enhancements

The hardware has been modified to allow the **IDN BRIDGE** software to pulse two pins on the maintenance connector.

The signals are those on the DCE maintenance connector pins 6 and 8.

Pin 6 is an RS-232-level signal driven at 5 Hz frequency during normal operation of the **IDN BRIDGE** application software. This signal is referred to as the Alarms Poll. When the bridge detects an alarming condition, it terminates the Alarms Poll. This triggers the external alarm unit to display an alarm condition and/or to trigger an audible alarm. Once the Alarms Poll has occurred, it may be reset via operator command. This signal is not driven during diagnostics, boot loader, NVM backup, or during the monitor prompt when in maintenance mode.

Pin 8 is an RS-232-level signal driven at 5 Hz frequency during normal operation of the **IDN BRIDGE** application software. This signal is referred to as the Live/Standby Poll and is regarded as a *heartbeat* into an external switching unit. The heartbeat ceases on a bridge failure and triggers a Live/Standby switch, thereby enabling the Standby bridge to establish connection to the remote partner. This signal is not driven during diagnostics, boot loader, NVM backup, or during the monitor prompt when in maintenance mode.

4.1.2 Software Enhancements

4.1.2.1 Live/Standby and Alarm Poll

When an **IDN BRIDGE** is started, it assumes a Standby status and generates the Live/Standby Poll (Live Signal) and Alarms Poll (Healthy Signal). If either WAN link becomes active then the **IDN BRIDGE** will transition to a Live status and continues with normal operation. If all WANs become inactive for more than 15 seconds the **IDN BRIDGE** will transition to a Standby status and terminate the Alarms Poll (Healthy Signal).

During normal (Live) operation the **IDN BRIDGE** will generate the Live/Standby Poll (Live Signal) and Alarms Poll (Healthy Signal). The following events will cause the **IDN BRIDGE** to terminate the Alarms Poll (Healthy Signal) but the Live/Standby Poll (Live Signal) will still be generated.

1. Exceeding the LAN frames deleted threshold. The **IDN BRIDGE** maintains a count of the frames missed or discarded due to receiver errors detected by the LAN interface chip. Errors of this type include CRC, framing and overflow errors. If the per second count (the counter is reset to zero every second) exceeds a configurable threshold then the **IDN BRIDGE** will terminate the Alarms Poll (Healthy Signal). The threshold is set using the **set monitor enet deleted threshold** operator command. Setting the threshold to zero will disable this feature. The counter is reset to zero every second.

2. Exceeding the Multilink frames lost threshold. The **IDN BRIDGE** maintains a count of the frames discarded due to exceeding queue thresholds and unavailable resources detected by the Multilink code. If the count exceeds a configurable threshold then the **IDN BRIDGE** will terminate the Alarms Poll (Healthy Signal). The threshold is set using the **set monitor mlink lost threshold** operator command. Setting the threshold to zero will disable this feature. The counter is reset to zero by the **clear monitor alarm** operator command or by the transition from Standby to Live status.

3. Exceeding the WAN receive error threshold. The **IDN BRIDGE** maintains a count of the frames missed or discarded due to receiver errors detected by the WAN interface chip. Errors of this type include non-octet aligned and FCS errors. The count is used to calculate a ratio of good frames received within the current sample period. If this ratio falls below a configurable threshold (expressed as a percentage) then the **IDN BRIDGE** will terminate the Alarms Poll (Healthy Signal). The threshold and sample period is set using the **set monitor lapb error threshold** operator command. Setting the threshold or sample period to zero will disable this feature. The error counter used to calculate the ratio is reset to zero at system startup and by the **clear monitor alarm** and **clear monitor stats** operator commands.

4. Detecting an inactive WAN. The **IDN BRIDGE** monitors the state of each WAN link. If the WAN link fails to transmit data (including the hello poll packets as configured via the **set monitor hello_poll** operator command) the link will transition from the Up to Down state. If either WAN link transitions from the Up to Down state, or if either WAN link fails to transition from Down to Up within 10 seconds of each other, then the **IDN BRIDGE** will terminate the Alarms Poll (Healthy Signal). Transition to the Down state include unrecoverable framing errors or excessive (T1*N2 LAPB retries) unsuccessful retransmission attempts due to physical link problems or disconnection. This feature can be disabled with the use of the **set monitor lapb inactivity** operator command.

The following events will cause the **IDN BRIDGE** to terminate both the Live/Standby Poll (Healthy Signal) and the Alarms Poll (Live Signal).

1. Exceeding the LAN transmit failure threshold. The **IDN BRIDGE** maintains a count of failed transmission attempts (including the hello poll packets as configured via the **set monitor hello_poll** operator command) due to transmitter errors detected by the LAN interface chip. Errors of this type include loss of carrier, late collisions, and excessive unsuccessful

retransmission attempts due to physical link problems or disconnection. If the count exceeds a configurable threshold then the **IDN BRIDGE** will terminate the Live/Standby Poll (Live Signal) and the Alarms Poll (Healthy Signal). The threshold is set using the **set monitor enet transmit threshold** operator command. Setting the threshold to zero will disable this feature. The counter is reset to zero after each successful transmission.

2. Exceeding the LAN receive interpacket time. The **IDN BRIDGE** starts a timer after successfully receiving a frame from the LAN link. If the timer exceeds a configurable period (expressed in seconds) then the **IDN BRIDGE** will terminate the Live/Standby Poll (Live Signal) and the Alarms Poll (Healthy Signal). The period is set using the **set monitor enet interpacket time** operator command. Setting the period to zero will disable this feature.

3. Set bridge standby. The **set bridge standby** operator command will cause the Live/Standby Poll (Live Signal) and the Alarms Poll (Healthy Signal) to be terminated. This command is provided to allow the operator to force a Live/Standby switch for maintenance purposes.

To restore the Live/Standby Poll (Live Signal) and Alarms Poll (Healthy Signal) the network administrator can issue the **clear monitor alarm** operator command.

4.1.2.2 Additional Trap Messages

The **IDN BRIDGE** will generate informational trap (security level 2) messages under the following conditions:

1. CPU Utilization threshold transition. The **IDN BRIDGE** will sample the CPU load 10 times per second. As the average value (expressed as a percentage over a 10-second period) transitions above or below the configurable threshold the **IDN BRIDGE** will generate the following trap messages during normal (Live) operation.

   ```
   CPU utilization [above|below] threshold
   ```

   The threshold is set using the **set monitor cpu_load threshold** operator command. Setting the threshold to zero will disable this feature.

2. WAN Transmit Utilization threshold transition. The **IDN BRIDGE** will calculate the WAN transmit utilization every 10 seconds. As the utilization (expressed as a percentage) transitions above or below the configurable threshold the **IDN BRIDGE** will generate the following trap message during normal (Live) operation.

   ```
   [J3|J3B] utilization [above|below]  threshold
   ```

   The threshold is set using the **set monitor lapb utilization percent** operator command. To determine the WAN transmit utilization the **IDN BRIDGE** must know the physical interface speed. Since the **IDN BRIDGE** cannot calculate the speed of an externally provided clock, the operator must set the speed using the **set monitor lapb utilization speed**

operator command. Setting the threshold or speed to zero will disable this feature.

3. Memory Utilization threshold transition. Since the purpose of the Memory Utilization Monitor is to filter low priority LAN frames, the **IDN BRIDGE** calculates the memory utilization for every frame received by the LAN. As the utilization (expressed as a percentage) transitions above the high threshold or below the low threshold, the **IDN BRIDGE** will generate the following trap message during normal (Live) operation.

```
MEM utilization [above|below] threshold
```

The threshold is set using the **set monitor buffer [high|low] threshold** operator command. Setting either threshold to zero will disable this feature.

Note*:* Since this calculation is performed for each LAN frame received, this feature has the potential of generating numerous trap messages. To avoid a high volume of MEM utilization trap messages the **IDN BRIDGE** will only generate the trap message on the first occurrence of the condition after a powerup, reset, reload, or after the **clear monitor stats** operator command or (Z)ero from the status monitor.

4. Alarm Status Transition. During normal (Live) operation, the **IDN BRIDGE** will generate one of the following trap message as the Alarm Status transitions from Off to On or from On to Off.

```
Bridge alarm on: WAN inactive

Bridge alarm on: WAN error threshold exceeded

Bridge alarm on: LAN frames deleted threshold exceeded

Bridge alarm on: LAN receive interpacket time exceeded

Bridge alarm on: LAN transmit error threshold exceeded

Bridge alarm on: Multilink frames lost threshold exceeded

Bridge alarm off:
```

5. Live/Standby Status Transition. The **IDN BRIDGE** will generate the following trap message as the Live/Standby Status transitions from Live to Standby or from Standby to Live.

```
Bridge Transition: [Live|Standby]
```

4.1.2.3 Enhanced FDB Range Specification

The user interface allows a single user command to modify the filtering database (FDB) by adding or deleting numerous entries. Commands have been added to add and delete a range of FDB entries. The user enters a command that specifies the starting and ending MAC (Media Access Control) addresses along with a disposition and port. The packet disposition and port parameters are identical to the existing **add fdb entry** and **delete fdb**

entry commands. The **add fdb entry**, **delete fdb entry, display fdb entry**, and **display fdb table** commands have not changed.

An algorithm exists that specifies how the MAC addresses should be determined within the user-specified range. The following section describes this algorithm. A MAC address has the form AA:BB:CC:DD:EE:FF when viewed within the FDB. The addresses within the range are computed by first incrementing BB with overflow, causing CC to be incremented. Overflow of the CC byte is lost.

Thus a range defined with the new commands by a starting address of 11:22:33:44:55:66 and an ending address of 11:78:34:44:55:66 includes the following entries in the FDB:

11:22:33:44:55:66
11:23:33:44:55:66
11:24:33:44:55:66
11:25:33:44:55:66
:
:
:
11:FD:33:44:55:66
11:FE:33:44:55:66
11:FF:33:44:55:66
11:00:34:44:55:66
11:01:34:44:55:66
11:02:34:44:55:66
:
:
:
11:76:34:44:55:66
11:77:34:44:55:66
11:78:34:44:55:66

Note: Individual entries in the FDB that were added as the result of the **add fdb range entry** command cannot be deleted by the **delete fdb entry** command.

Formats of the new commands to add and delete FDB entries using the range mechanism are in Section 7.

All commands described in Section 7 have been added to the **IDN BRIDGE** command set. New objects have been added to the SNMP MIB that are specific to this enhanced product.

4.1.2.4 Multi-Page Continuous Status Monitor

The Status Monitor facility permits a continuous display of various status indications on the locally attached maintenance terminal.

When the **Status Monitor** is entered via a user command, the status monitor periodically (interval is configured via user command) updates the maintenance terminal screen with the current values of various status indicators and counters. When the Status Monitor is executing, the console user cannot enter standard **IDN BRIDGE** line commands. The

only commands accepted during this time are those specified by the Status Monitor (one of which is to exit the Status Monitor and return to normal command line mode processing).

The default operating mode of the Status Monitor does not make use of any cursor addressing or graphical features of the attached console terminal. Thus a standard dumb terminal (at least 20 lines and 80 columns) can still be used. Commands entered to the Status Monitor are single letter commands.

Note: The Status Monitor provides access to status indications and counters of the local system only. It does not display the status of system variables of a remote system.

Note: The Status Monitor is inherently restricted to a maximum of four LAPB interfaces in a single Multilink group, and the first Ethernet interface encountered.

The Status Monitor consists of multiple pages. One page will be dedicated to displaying the Thresholds, Statistics, and Error counters. A second page will be dedicated to displaying various Alarm States.

The following is the screen layout for statistics and error counters:

```
WAN Port/Status               J3.1/DOWN   J3.2/SETUP  J4.1/UP     J4.2/STDBY
WAN Inactivity Flag/Alarm       ON/OFF      ON/OFF      ON/OFF      ON/OFF
WAN Error Thresh/Ratio          97/100      97/100      97/100      97/100
WAN Tx Utilization Thresh       66          66          66          66
WAN Tx Utilization Stats(C/H)    0/95 *      0/95 *      0/95 *      0/95 *
WAN Transit Delay (msec)        48          39          48          140
WAN Ave Opkt Size              186         167         161         185

LAN Port                        J1
LAN Frames Deleted Thresh        1       Alarm Status OFF
LAN Tx Errors Thresh            10       Alarm Status OFF
LAN Rx Interpacket Time          0       Alarm Status OFF
LAN Ave Ipkt Size              256

Multilink Frames Lost Thresh     1       Count          0
CPU Utilization Thresh          66       Stats(C/H)    0/50
MEM Utilization Thresh(L/H)    80/90     Stats(C/H)    0/13      Disc     0

Operational Status              LIVE     Alarm Status  ON
                  (P)age, (U)pdate, (C)lear Alarms, (Z)ero, (E)xit
TRAP line 1
TRAP line 2
```

`WAN Port/Status`: The `WAN Port/Status` line indicates the IDN BRIDGE port identifier (J3.1, J4.1, etc) and the current connection status of the LAPB link. The connection status will indicate one of the following:

`DOWN` the Transit Timer TEST message or the LAPB protocol handshake was not successful or is currently in-progress.

`SETUP` the Transit Timer TEST message exchange was successful, the LAPB protocol handshake was successful, and the link is in the process of becoming a participant of the Multilink Group.

UP the Transit Timer TEST message exchange was successful, the LAPB protocol handshake was successful, and the link is an active participant of the Multilink Group.

STDBY the Transit Timer TEST message exchange was successful, the LAPB protocol handshake was successful, but the link is an inactive member of the Multilink Group due to the measured transit delay.

WAN Inactivity Flag/Alarm: The `WAN Inactivity Flag` indicates **on** when the IDN BRIDGE is monitoring the LAPB link for inactivity, and **off** when it is not. This parameter is configured by the `set monitor lapb inactivity` operator command. The `WAN Inactivity Alarm` indicates **on** when the IDN BRIDGE is monitoring the LAPB link for inactivity and has detected an inactive link.

WAN Error Thresh/Ratio: The `WAN Error Thresh` indicates the alarm threshold for the WAN receiver error ratio. This parameter is configured by the `set monitor lapb error threshold` operator command. The `WAN Ratio` indicates the percentage of good frames within the current frame sample period.

WAN Tx Utilization Thresh: The `WAN Utilization Thresh` indicates the alarm threshold for the WAN transmit utilization percentage. This parameter is configured by the `set monitor lapb utilization percent` operator command.

WAN Tx Utilization Stats(C/H): The `WAN Utilization Stats` indicates the **current/highest** WAN utilization percent as calculated every 10 seconds by the IDN BRIDGE.

WAN Transit Delay (msec): The `WAN Transit Delay` indicates the amount of time (expressed in milli-seconds) required for the IDN BRIDGE to transmit a 1520 byte test over the WAN link.

WAN Ave Opkt Size: The `WAN Ave Opkt Size` indicates the average frame size (expressed in bytes) transmitted by the IDN BRIDGE over the WAN link during the past 10 second period

LAN Port: The `LAN Port` line indicates the IDN BRIDGE port identifier **J1**.

LAN Frames Deleted Thresh: The `LAN Frames Deleted Thresh` indicates the alarm threshold for the LAN receiver error counter. This parameter is configured by the `set monitor enet deleted threshold` operator command.

LAN Tx Errors Thresh: The `LAN Tx Errors Thresh` indicates the alarm threshold for the LAN transmit error counter. This parameter is configured by the `set monitor enet transmit threshold` operator command.

`LAN Rx Interpacket Time`: The `LAN Rx Interpacket Time` indicates the maximum amount of time (expressed in seconds) the LAN may be idle before the IDN BRIDGE triggers an alarm condition. This parameter is configured by the `set monitor enet interpacket time` operator command.

`LAN Ave Ipkt Size`: The `LAN Ave Ipkt Size` indicates the average frame size (expressed in bytes) received by the IDN BRIDGE over the LAN link during the past 10 second period.

`Multilink Frames Lost Thresh`: The `Multilink Frames Lost Thresh` indicates the alarm threshold for the Multilink frames lost error counter. This parameter is configured by the `set monitor mlink lost threshold` operator command.

`Multilink Frames Lost Count`: The `Multilink Frames Lost Count` indicates the number of frames discarded due to exceeding queue thresholds and unavailable resources detected by the Multilink code.

`CPU Utilization Thresh`: The `CPU Utilization Thresh` indicates the alarm threshold for the CPU utilization percentage. This parameter is configured by the `set monitor cpu_load threshold` operator command.

`CPU Utilization Stats(C/H)`: The `CPU Utilization Stats` indicates the **`current/highest`** CPU utilization percent as calculated by the IDN BRIDGE.

`MEM Utilization Thresh(H/L)`: The `MEM Utilization Thresh` indicates the **`high/low`** alarm thresholds for the memory utilization percentage. This parameter is configured by the `set monitor buffer [high|low] threshold` operator command.

`MEM Utilization Stats(C/H)`: The `MEM Utilization Stats` indicates the **`current/highest`** memory utilization percent as calculated by the IDN BRIDGE.

`MEM Utilization Disc`: The `MEM Utilization Disc` indicates the number of low priority frames discarded due to exceeding the memory utilization threshold.

`Operational Status`: The `Operational Status` indicates either **`LIVE`** or **`STANDBY`** as conditions warrant.

`Alarm Status`: The `Alarm Status` indicates either **`ON`** or **`OFF`** as conditions warrant.

Commands entered to the Status Monitor are single letter commands. Valid commands to the Status Monitor include the following:

`(P)age` – toggle to next page of Status Monitor.

`(U)pdate` - to update the screen with the current values. This command is provided so that the administrator can manually invoke a screen update.

(C)lear - to clear alarm condition (similar to `clear monitor alarm` command).

(Z)ero - to zero the statistic counters (similar to `clear monitor stats` command).

(E)xit - to exit the Status Monitor and return to normal command line processing.

The following is the screen layout for warnings and alarms:

```
                                    J3.1    J3.2    J4.1    J4.2    CPU     Memory
Utilization Warnings                OFF     OFF     OFF     OFF     OFF     OFF

                                    J3.1    J3.2    J4.1    J4.2
WAN Inactivity Alarm                OFF     OFF     OFF     OFF
WAN Error Thresh Alarm              OFF     OFF     OFF     OFF

                                    J1
LAN Frames Deleted Thresh Alarm     OFF
LAN Tx Error Thresh Alarm           OFF
LAN Rx Interpacket Time Alarm       OFF

MLINK Frames Lost Thresh Alarm      OFF

Overall Alarm Status                OFF

Operational Status                  LIVE

                    (P)age, (U)pdate, (C)lear Alarms, (Z)ero, (E)xit

TRAP line 1
TRAP line 2
```

4.1.2.5 HTML Status/Error Monitor Pages

New pages have been defined which allow access to all Reuters specific commands via HTML including the Status/Error monitor. This HTML Status/Error Monitor provides a manual trigger to start the continuous update process, update for a configurable period (default of 15 minutes), and then stops.

4.1.2.6 Hello Polling

A new feature has been added which provides a better “link-down” detection on idle links. A new command has been defined to allow the operator to enable and disable this feature on an individual link basis. If this feature is enabled for a specified interface then the interface will periodically generate and transmit a “hello” packet if nothing has been transmitted during the configured period.

4.1.2.7 Packet Prioritization

Received packets from the Ethernet are grouped into two priorities, low and high. All packets have a low priority associated with them by default. High priority is assigned to packets with specific source and/or destination addresses. When the user-configured buffer high-water threshold (measured as a percentage of total available buffers) is exceeded, low priority received Ethernet frames are discarded. Low priority Ethernet frames will continue to be discarded until the user-configured buffer low-water threshold

is reached. It is the responsibility of the end system to guarantee delivery of low priority packets if required by the application.

Packets are discarded prior to having Multilink headers applied. Therefore Multilink sequence errors will not occur at the receiving bridge. The Multilink process will first determine which port the packet is to be transmitted out and then determine if that port is in a state where the low priority packets are to be discarded.

Packets received from the Ethernet are processed by filtering the destination MAC address with all entries in the Forwarding database. Packets, which are not discarded during this process, are then checked against each of entry in the Destination Address Priority Range Table to determine if the packet should have a high priority associated with it. If not, each entry in the Source Address Priority Range table is checked to determine if the source MAC address of the received packet matches one of the entries in this table. If so, high priority is associated with the packet.

Note: The existing protocol type filtering is not affected by this modification.

4.1.2.8 Automatic T1 Timer Calculation

When the **IDN BRIDGE** sends a packet out the WAN link it expects a response (acknowledgment) within the time period set by the T1 Timer. The T1 timer value must be large enough to send the maximum size frame but small enough to quickly detect link failures. The maximum frame size, link speed, and delays introduced by satellite hops must all be considered when choosing the optimum value for the T1 Timer.

The **IDN BRIDGE** will automatically calculate and set the optimum T1 Timer value for each WAN link. Before each WAN link is established (LAPB protocol handshake) a 1520 byte test message is transmitted and echoed back by each end of the link. The test message is periodically re-transmitted until a successful echo is received. The test message re-transmission period is configurable using the **set monitor lapb testtmo** operator command. The WAN transit delay (measured in milli-seconds) is half the round-trip time (assuming equal send/received delays). The T1 Timer is automatically set to the transit delay measurement multiplied by a configurable transit delay multiplier and then a configurable T1 Timer variance is added. The multiplier is set using the **set monitor lapb transit_multi** operator command. The variance is set using the **set monitor lapb t1 variance** operator command

If the difference in delays between any two WAN links (using the delay values measured by the auto T1 timer calculation) exceeds a user configurable value (expressed in milliseconds) then the slower (longest delay) of the two links will be put into a standby state. The user configurable delay delta is set using the **set monitor mlink delay delta** operator command. While a link is in the standby state it will not transmit/forward any data. Every link transition will cause the standby status for all links to be re-evaluated.

4.1.2.9 Automatic Configuration

A new command has been added to the standard command set which allows an operator to issue a single command to pre-configure various operating parameters specific to the Reuters IDN environment. The new command, **set bridge type**, will allow the operator to

configure the **IDN BRIDGE** as either a North or South type bridge. This command will invoke a script, which issue a list of commands specific to the North or South bridge type. As each command is issued it will be echoed to the locally attached operator terminal. The responses to each command will be suppressed except in the case of an error. An error can occur when the **set bridge type** command is issued before NVM has been initialized.

4.1.2.10 Multilink Frame Packing

Frame packing provides more efficient use of WAN bandwidth by reducing the number of LAPB frames thus reducing the LAPB protocol overhead (Address, control and FCS). By reducing the number of LAPB frames more data can be sent before saturating the LAPB frame window (the number of frames that can be sent before requiring acknowledgment of receipt of the frame) and the CPU is interrupted less frequently in sending and receiving LAPB frames.

When frame packing is enabled via the **set monitor mlink packing type** operator command all frames received from the LAN interface which are to be forwarded over the WAN interface will be packed into one large buffer. Frames received from the LAN interface which are not being forwarded (such as SNMP management traffic which terminates at the bridge) do not get packed. If the size of the LAN frame would cause the packed buffer to exceed 1520 bytes then the **IDN BRIDGE** will transmit the existing packed frame out the WAN interface and create a new packed buffer. If the **IDN BRIDGE** does not receive a frame from the LAN interface within a user configurable time-out period then it will transmit the existing packed frame The time-out period is configured by the **set monitor mlink packing tmo** operator command.

4.1.2.11 Multilink Frame Recovery

Enhancements have been made to allow the recovery of Multilink Frames that have been compressed and queued for transmission by the LAPB module (queued on the HPC DMA rings) in the event of physical link failures. The Multilink Frame Recovery enhancement provides a more robust mechanism for the delivery of frames. This is especially important for applications (such as in the Reuters IDN) where the use of upper level protocols (such as the TCP/IP stack) is not relied upon for guaranteed delivery of frames. When a link failure is detected, the frames that are queued by the LAPB protocol module will be de-queued from the failed links queue and sent back to the Multilink Protocol Module for re-transmission (without decompressing or re-sequencing).

4.2 New Capabilities

There are no new capabilities. All existing capabilities of the Amazon Rapid hardware and the 11.1.13.7 software are being merged or migrated into the ACX 470 hardware and 12.0 software.

4.3 Backward Compatibility

The AXC 470 IDN and Reuters Enhanced 12.0 software baseline will be compatible (will reliably establish WAN connectivity) with the Amazon Rapid and 11.1.13.7 software.

4.4 Forward Extensibility

The Reuters Enhanced 12.0 software baseline will not be compatible (will not reliably establish WAN connectivity) with the other bridge vendors due to the proprietary nature of the Multilink Frame Recovery feature. The Reuters Enhanced 12.0 software baseline will not be compatible with the Standard 12.0 software baseline. This is mainly due to the additional information embedded in the Multilink Protocol header for supporting the Multilink Frame Recovery feature.

5 Managed Objects

5.1 Reuters Specific Objects

The input to the CLISPEC utility best describes the Reuters Specific Objects that are managed by this product. See section A.1

6 Managed Groups

6.1 Reuters Specific Groups

The input to the CLISPEC utility best describes the Reuters Specific Groups that are managed by this product. See section A.1

7 Interfaces

7.1 Command Line Interface

7.1.1 Clear monitor alarm

Function

This command clears the alarm condition for the **IDN BRIDGE**.

Format

```
clear monitor alarm
```

Usage

All conditions that would cause an alarm are cleared.

The output of this command is as follows:

```
Clear Alarm  = OFF
```

7.1.2 Clear monitor stats

Function

This command clears the status monitor statistics counters in an **IDN BRIDGE** system. This includes the WAN Tx utilization statistics counter, CPU utilization statistics counter, and the memory utilization statistics counter.

Format

```
clear monitor stats
```

Usage

The output of this command is as follows:

```
Clear Stats  = 2
```

7.1.3 Set bridge standby

Function

This command is used to force the **IDN BRIDGE** into standby mode such that maintenance can be performed on the bridge. This command will cause the Live and Healthy 5HZ signals to cease.

Format

`set bridge standby`

Usage

The output of this command is as follows:

```
BrStat = 2
```

7.1.4 Set bridge type

Function

This command is used to invoke the automatic configuration of the **IDN BRIDGE** as either a NORTH or SOUTH type bridge.

Format

`set bridge type` [**`none`**|**`north`**|**`south`**] *{count}*

Usage

Parameters and variables are defined as follows:

- **`NORTH`** and **`SOUTH`** are the options for the bridge type configuration

  `NORTH` will invoke the command set specific for a North type **IDN BRIDGE**.

  `SOUTH` will invoke the command set specific for a South type **IDN BRIDGE**.

  The default value is **`NONE`.**

- *count* specifies the number of LAPB interfaces to be incorporated into the Multilink Group. This parameter is useful when 4 WAN interfaces are installed, but only two are actually connected.

This command takes effect immediately and should only be issued after NVM has been initialized. The commands are echoed as the command set is issued, but the responses are suppressed unless there is an error. Errors can happen if NVM is not previously initialized.

The following is the list of commands that get executed when the operator issues the "SET BRIDGE TYPE NORTH 2" command:

```
set prompt North
set lapb station type J3.1 dce
set lapb station type J3.2 dce
set physical port protocol J3.1 lapb
set physical port protocol J3.2 lapb
set lapb clock mode J3.1 slave
set lapb clock mode J3.2 slave
add multilink group entry m1
add multilink group physical port m1 J3.1
add multilink group physical port m1 J3.2
set bridge port status 2 disable
set bridge port status 3 disable
set bridge port status 4 disable
set bridge port status 5 disable
set bridge port status 6 disable
add bridge port entry multilink 7 m1
set physical port compression J3.1 on 1
set physical port compression J3.2 on 1
set bridge port status 7 enable
set multilink group admin status m1 enable
add protocol priority entry 1 0x0800 discard
add protocol priority entry 1 0x0806 discard
add protocol priority entry 1 0x6001 discard
add protocol priority entry 1 0x6002 discard
add protocol priority entry 1 0x6003 discard
add protocol priority entry 1 0x6004 discard
add protocol priority entry 2 0x6006 discard
add protocol priority entry 1 0x6007 discard
set monitor mlink lost threshold J3.1 1
set monitor mlink lost threshold J3.2 1
set monitor lapb utilization percent J3.1 66
set monitor lapb utilization percent J3.2 66
set monitor lapb inactivity J3.1 on
set monitor lapb inactivity J3.2 on
set monitor enet deleted threshold J1 1
set monitor enet transmit_errors threshold J1 10
set monitor enet interpacket time J1 0
set monitor cpu threshold 66
set monitor buffer high threshold 90
set monitor buffer low threshold 80
add fdb destination priority range entry ff:00:00:ff:ff:ff ff:fe:ff:ff:ff:ff
add fdb source priority range entry 00:f7:ff:ff:ff:ff 00:fe:ff:ff:ff:ff
add fdb source priority range entry 00:f5:ff:ff:ff:ff 00:f6:ff:ff:ff:ff
```

The following is the list of commands that get executed when the operator issues the "SET BRIDGE TYPE SOUTH 2" command:

```
set prompt South
set lapb station type J3.1 dte
set lapb station type J3.2 dte
set physical port protocol J3.1 lapb
set physical port protocol J3.2 lapb
set lapb clock mode J3.1 slave
set lapb clock mode J3.2 slave
add multilink group entry m1
add multilink group physical port m1 J3.1
add multilink group physical port m1 J3.2
set bridge port status 2 disable
set bridge port status 3 disable
set bridge port status 4 disable
set bridge port status 5 disable
set bridge port status 6 disable
add bridge port entry multilink 7 m1
set physical port compression J3.1 on 1
set physical port compression J3.2 on 1
set bridge port status 7 enable
set multilink group admin status m1 enable
```

```
add protocol priority entry 1 0x0800 discard
add protocol priority entry 1 0x0806 discard
add protocol priority entry 1 0x6001 discard
add protocol priority entry 1 0x6002 discard
add protocol priority entry 1 0x6003 discard
add protocol priority entry 1 0x6004 discard
add protocol priority entry 2 0x6006 discard
add protocol priority entry 1 0x6007 discard
set monitor mlink lost threshold J3.1 1
set monitor mlink lost threshold J3.2 1
set monitor lapb utilization percent J3.1 66
set monitor lapb utilization percent J3.2 66
set monitor lapb inactivity J3.1 on
set monitor lapb inactivity J3.2 on
set monitor enet deleted threshold J1 1
set monitor enet transmit_errors threshold J1 10
set monitor enet interpacket time J1 0
set monitor cpu threshold 66
```

7.1.5 Set monitor buffer high threshold

Function

This command is used to specify the high water threshold for memory utilization monitoring within the system.

Format

`set monitor buffer high threshold` *`{high_water}`*

Usage

- *high_water* specifies the maximum percentage of the total number of buffers which can be in use at a given time. A high or low water threshold of 0 effectively disables the monitoring of memory utilization. When the high water threshold is exceeded, low priority Ethernet frames will be discarded when received. Frames already received and queued will not be affected. The allowable range for this parameter is 0 through 100.

The low priority Ethernet frames will continue to be discarded until the total number of buffers in use falls below the l*ow_water* threshold. The *low_water* mark is specified as percentage of the total number of buffers that are in use at a given time.

This command takes effect after the next system reset. An **IDN BRIDGE** is preset with a buffer high water threshold of 0 (disabled.)

7.1.6 Set monitor buffer low threshold

Function

This command is used to specify the low water threshold for memory utilization threshold monitoring within the system.

Format

`set monitor buffer low threshold` *`{low-water}`*

Usage

- The *low_water* threshold is used by the memory monitoring function of the bridge to determine when low priority Ethernet frames will no longer be discarded. The *low_water* threshold is specified as a percentage of the total number of buffers that are in use at a given time. A high or low water threshold of 0 effectively disables the monitoring of memory utilization. The allowable range for this parameter is 0 through 100.

This command takes effect after the next system reset. An **IDN BRIDGE** is preset with a buffer low water threshold of 0 (disabled.)

7.1.7 Set monitor cpu_load

Function

This command is used to specify the CPU utilization threshold for status monitoring functionality.

Format

`set monitor cpu_load threshold` {*`percent`*}

Usage

- *percent* specifies the percentage of CPU utilization which when exceeded will cause generate a TRAP warning message. The allowable range of this parameter is 0 through 100. A value of zero (0) disables this function. The preset value is zero (0).

This command takes effect at the next CPU utilization calculation period (every 10 seconds).

7.1.8 Set monitor enet deleted threshold

Function

This command is used to specify the number of Ethernet frames that can be deleted before an alarm condition occurs.

Format

`set monitor enet deleted threshold` *{port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of Ethernet frames that must be lost (during a 1 second interval) to cause an alarm condition. The valid range of this parameter is 0 to 65535. A value of zero (0) disables this function. (All values greater than the capacity of the Ethernet effectively disable this function.) The preset value is zero (0).

A **reset** command must be issued for this command to take effect.

7.1.9 Set monitor enet interpacket time

Function

This command is used to specify the maximum time between the reception of frames on the specified Ethernet interface.

Format

`set monitor enet interpacket time` *{port_id} {seconds}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *seconds* specifies the maximum number of seconds between reception of frames on the specified Ethernet interface. If a frame is not received in this period of time or the time between the reception of two frames on the specified Ethernet interfaces exceeds this time interval, a Live/Standby condition occurs. The valid range of this parameter is 0, 10 to 6000. A value of zero (0) disables this function. The preset value is zero (0).

A **reset** command must be issued for this command to take effect.

7.1.10 Set monitor enet transmit_errors threshold

Function

This command is used to specify the number of consecutive Ethernet frame transmissions resulting in an error that caused a Live/Standby condition to occur.

Format

`set monitor enet transmit_errors threshold` *{port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of consecutive Ethernet frame transmissions resulting in an error to cause a Live/Standby condition. The valid range of this parameter is 0 to 65535. A value of zero (0) disables this function. The preset value is zero (0).

A **reset** command must be issued for this command to take effect.

7.1.11 Set monitor hello_poll

Function

This command is used to specify when the hello poll interval for the specified interface. If the interface has not transmitted data within this interval then a hello poll message will be transmitted.

Format

`set monitor hello_poll` {*port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the hello poll interval in seconds. The allowable range of this value is from 0 to 65535 seconds. The value of zero disables this function. The preset value is zero.

This command takes effect immediately.

7.1.12 Set monitor LAPB error threshold

Function

This command is used to specify the error threshold of the specified LAPB serial line for alarm monitoring.

Format

`set monitor lapb error threshold` *{port_id} {percent} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *percent* specifies the ratio of good frames to total frames received which must be maintained to avoid an alarm condition. The valid range for this parameter is 0 to 100. A value of zero (0) disables this function. The preset value is 97.
- *count* specifies the number of frames which must be received before evaluating the LAPB error ratio. The valid range for this parameter is 0 to 65535. A value of zero (0) disables this function. The preset value is 500.

A **reset** command must be issued for this command to take effect.

7.1.13 Set monitor LAPB inactivity

Function

This command is used to enable/disable the monitoring of LAPB link inactivity alarms.

Format

`set monitor LAPB inactivity` *{port_id}* [**`on`**|**`off`**]

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- **on** or **off** are the options for LAPB inactivity monitoring

  on enables the monitoring of LAPB link inactivity alarms.
  off disables the monitoring of LAPB link inactivity alarms.

The default value is **on**. This command takes effect immediately.

7.1.14 Set monitor LAPB transit_multi

Function

When a LAPB link is established a transit delay for the link is measured. The transit delay is the minimum amount of time (measured in milli-seconds) required to transmit a 1520 byte test message.

This command is used to specify a multiplier to the LAPB transit delay measurement used for the automatic calculation of the LAPB T1 timer.

Format

`set monitor lapb transit_multi` *`{port_id} {count}`*

Usage

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the multiplication factor to the transit delay measurement used for calculating the LAPB T1 timer. The allowable range of this parameter is 2 through 16000. The preset value is two (2).

This command takes effect immediately.

7.1.15 Set monitor LAPB testtmo

Function

When a LAPB link is established a test message is transmitted for the purpose of measuring the transit delay. If a response is not received, the test message is re-transmitted until a response is received.

This command is used to specify how frequent the test message is re-transmitted.

Format

`set monitor lapb testtmo` *`{port_id} {count}`*

Usage

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the period between the transmission of successive test messages. The allowable range for this parameter is 10 through 600 tenths of a second. The preset value is 15 (1.5 seconds).

This command takes effect immediately.

7.1.16 Set monitor LAPB utilization percent

Function

This command is used to specify the transmit utilization percentage of the specified LAPB serial line for status monitoring functionality.

Format

`set monitor lapb utilization percent` *`{port_id} {percent}`*

Usage

- *port_id* identifies the **IDN BRIDGE** port.
- *percent* specifies the percentage of utilization of the LAPB serial line when exceeded will cause an indication to the user. This can be used to determine if the capacity of the LAPB serial line is sufficient for the user's application. This allowable range for this parameter is 0 through 100.

This command takes effect after the next system reset. An **IDN BRIDGE** is preset with a *percent* of 0.

A *LAPB utilization speed* of 0 or a *LAPB utilization percent* of 0 disables the monitoring of the LAPB utilization.

7.1.17 Set monitor LAPB utilization speed

Function

This command is used to specify the speed of the specified LAPB serial line for status monitoring functionality computations.

Format

`set monitor lapb utilization speed` *`{port_id} {line_speed}`*

Usage

- *port_id* identifies the **IDN BRIDGE** port.
- *line_speed* specifies the serial transmit clock rate of the specified port. Since the **IDN BRIDGE** cannot determine the speed of an externally provided serial transmit clock, this parameter must be provided by the administrator. The value set by the SET PHYSICAL PORT SPEED command is not used because this command cannot be used to set the all of the possible baud rates that may be supplied by an external clock source. The allowable range for this parameter is 0 to 2,048,000,000.

This command takes effect after the next system reset. An **IDN BRIDGE** is preset with a *line_speed* of 0.

A *LAPB utilization speed* of 0 or a *LAPB utilization percent* of 0 disables the monitoring of the LAPB utilization.

7.1.18 Set monitor LAPB 2lq

Function

This command is used to specify when the second level queuing (2lq) for the LAPB interface is invoked. This command is particularly useful in a Multilink environment to help improve performance when there is a large difference in transmission speed/delays in each LAPB member of a Multilink group.

Format

`set monitor lapb 2lq` *{port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of bytes which must be queued (on the primary LAPB transmit queue) for transmission over the LAPB interface before secondary queuing (Multilink queuing) is invoked. The valid range for this parameter is 0 to 65535. A value of zero disables the secondary queuing function. The preset value is zero (0).

This command takes effect immediately.

7.1.19 Set monitor LAPB t1 variance

Function

This command is used to specify a variance to be added to the LAPB T1 timer.

Format

`set monitor lapb t1 variance` *{port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of milli-seconds to be added to the LAPB T1 timer. The valid range for this parameter is 0 to 65535. The preset value is zero (0).

This command takes effect immediately.

7.1.20 Set monitor MLINK lost threshold

Function

This command is used to specify the threshold of the number of frames that may be lost during Multilink operation for alarm monitoring to function.

Format

`set monitor mlink lost threshold` *{port_id} {count}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of lost Multilink frames (since the last system restart or operator command to clear this counter) to cause an alarm condition. The valid range for this parameter is 0 to 65535. A value of zero disables this function. The preset value is zero (0).

This command takes effect immediately.

Note: Although this command requires a port_id, this command affects a system-wide variable and thereby the last command to set this parameter is the one used by the system.

7.1.21 Set monitor MLINK packing tmo

Function

This command is used to specify the frame packing time-out value.

Format

`set monitor mlink packing tmo` {*`port_id} {count}`*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the number of milli-seconds to wait for more data to pack. The valid range for this parameter is 0 to 65535. The preset value is ten (10) milli-seconds. The value of zero will cause the Multilink module to use the LAPB transit delay (which is calculated during LAPB link initialization) as the frame packing time-out value.

This command takes effect immediately.

Note: Although this command requires a port_id, this command affects a system-wide variable and thereby the last command to set this parameter is the one used by the system.

7.1.22 Set monitor MLINK packing type

Function

This command is used to specify the type of frame packing to be used.

Format

`set monitor mlink packing type` {*`port_id} {type}`*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *type* specifies the type of frame packing to be used. The valid range for this parameter is 0 to 2. The value 0 (the default) is for off (no frame packing). Two different frame-packing algorithms can be selected. The value of 1 will select the buffer copy algorithm. This algorithm is typically mode CPU intensive but utilizes less memory. The value of 2 will select the buffer-chaining algorithm. This algorithm is typically more memory intensive and less CPU intensive.

If frame packing is turned on then all frames received from the Ethernet interface which are to be forwarded over the WAN (via Multilink) will be packed into one large (1520 byte) buffer. Frames received from the Ethernet interface which are not being forwarded (such as SNMP management traffic which terminates at the **IDN BRIDGE**) do not get packed. The packing occurs at the Multilink level before Multilink sequence numbers are assigned and before compression is executed (if enabled). The Multilink module will look at each frame it receives from the Ethernet interface. If the size of the frame would cause the packed buffer to exceed 1520 bytes then the Multilink module will send the existing packed buffer and create a new packed buffer. If the Multilink module does not receive a frame from the Ethernet interface within a user configurable time-out period then it will send the partially packed frame. This command takes effect immediately.

Note: Although this command requires a port_id, this command affects a system-wide variable and thereby the last command to set this parameter is the one used by the system.

7.1.23 Set monitor MLINK delay delta

Function

This command is used to specify the minimum acceptable transit delay difference. If the difference in delays between any two WAN links (using the transit delay values measured by the auto T1 timer calculation) exceeds this user configurable value (expressed in milliseconds) then the slower (longest delay) of the two links will be put into a standby state. While a link is in the standby state it will not transmit/forward any data. Every link transition cause the standby status for all links to be re-evaluated

Format

`set monitor mlink delay delta` {*`port_id} {count}`*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the **IDN BRIDGE** port.
- *count* specifies the transit delay difference expressed in milli-seconds. The valid range for this parameter is 0 to 65535. A value of zero disables this function. The preset value is zero (0).

This command takes effect immediately.

Note: Although this command requires a port_id, this command affects a system-wide variable and thereby the last command to set this parameter is the one used by the system.

7.1.24 Set monitor status

Function

This command is used to enable/disable the continuous status monitoring feature.

Format

```
set monitor status [on|off|ansi]
```

Usage

- **on** specifies to enable the Status Monitor.
- **off** specifies to disable the Status Monitor.
- **ansi** specifies to enable the Status Monitor using ANSI cursor positioning escape sequences.

When the Status Monitor is executing, no commands can be entered by the user at the console terminal. However, a remote user can login to the system and issue this command to disable the Status Monitor. The Status Monitor can be disabled by the console user by entering the **exit** command to the Status Monitor. This command can only be issued to the local system.

This command takes effect immediately and a flag is saved in non-volatile memory so that the status monitor can be resumed after a system reset.

7.1.25 Display monitor status table

Function

This command will display a table of the current alarm condition status and live/standby status of the interfaces in an **IDN BRIDGE** system.

Format

```
display monitor status table
```

Usage

A sample output of this command appears as follows:

```
Port = J1                WAN Inactivity = N/A      WAN Error = N/A
WAN U-Current = 0        WAN U-Highest = 0         WAN T-Delay = 0
LAN Deleted = OFF        LAN Tx Error = OFF        LAN I-Packet Time = OFF
WAN Multilink Lost = N/A Avg Pkt Siz = 0           WAN State = DOWN

Port = J2                WAN Inactivity = N/A      WAN Error = N/A
WAN U-Current = 0        WAN U-Highest = 0         WAN T-Delay = 0
LAN Deleted = OFF        LAN Tx Error = OFF        LAN I-Packet Time = OFF
WAN Multilink Lost = N/A Avg Pkt Siz = 0           WAN State = DOWN

Port = J3.1              WAN Inactivity = OFF      WAN Error = OFF
WAN U-Current = 0        WAN U-Highest = 0         WAN T-Delay = 0
```

```
LAN Deleted = N/A          LAN Tx Error = N/A        LAN I-Packet Time = N/A
WAN Multilink Lost = OFF   Avg Pkt Siz = 0           WAN State = DOWN

Port = J3.2                WAN Inactivity = OFF      WAN Error = OFF
WAN U-Current = 0          WAN U-Highest = 0         WAN T-Delay = 0
LAN Deleted = N/A          LAN Tx Error = N/A        LAN I-Packet Time = N/A
WAN Multilink Lost = OFF   Avg Pkt Siz = 0           WAN State = DOWN
```

7.1.26 Display monitor parameter table

Function

This command will display a table of the currently configured monitoring parameters.

Format

`display monitor parameter table`

Usage

A sample output of this command will appear as follows:

```
Port = J1                  WAN Inactivity = OFF      WAN Err Threshold = 0
WAN Err Sample = 0         WAN U-Thresh = 0          WAN U-Speed = 0
WAN T-Multi = 0            WAN 2LQ = 0               WAN Test TMO = 0
LAN Deleted = 0            LAN Tx Error = 0          LAN I-Packet Time = 0
WAN Multilink Lost = 0     Delay Delta = 0           Packing Type = 0
Packing TMO = 10           Hello Poll = 0            T1 Variance = 0

Port = J2                  WAN Inactivity = OFF      WAN Err Threshold = 0
WAN Err Sample = 0         WAN U-Thresh = 0          WAN U-Speed = 0
WAN T-Multi = 0            WAN 2LQ = 0               WAN Test TMO = 0
LAN Deleted = 0            LAN Tx Error = 0          LAN I-Packet Time = 0
WAN Multilink Lost = 0     Delay Delta = 0           Packing Type = 0
Packing TMO = 10           Hello Poll = 0            T1 Variance = 0

Port = J3.1                WAN Inactivity = OFF      WAN Err Threshold = 97
WAN Err Sample = 500       WAN U-Thresh = 75         WAN U-Speed = 512000
WAN T-Multi = 2            WAN 2LQ = 0               WAN Test TMO = 15
LAN Deleted = 0            LAN Tx Error = 0          LAN I-Packet Time = 0
WAN Multilink Lost = 0     Delay Delta = 0           Packing Type = 0
Packing TMO = 10           Hello Poll = 0            T1 Variance = 0

Port = J3.2                WAN Inactivity = OFF      WAN Err Threshold = 97
WAN Err Sample = 500       WAN U-Thresh = 75         WAN U-Speed = 512000
WAN T-Multi = 2            WAN 2LQ = 0               WAN Test TMO = 15
LAN Deleted = 0            LAN Tx Error = 0          LAN I-Packet Time = 0
WAN Multilink Lost = 0     Delay Delta = 0           Packing Type = 0
Packing TMO = 10           Hello Poll = 0            T1 Variance = 0
```

7.1.27 Display monitor system info

Function

This command displays the current system alarm, live/standby, CPU load, and memory load parameters and status information.

Format

`display monitor system info`

Usage

A sample output of this command appears as follows:

```
Alarm Status = Off
Operational status = STANDBY
MEM U-Low Thresh = 55
MEM U-High Thresh = 85
MEM U-Current = 0
MEM U-Highest = 0
Low Prior Disc = 0
CPU U-thresh = 60
CPU U-Current = 0
CPU U-Highest = 0
```

7.1.28 Set monitor update interval

Function

This command is used to modify the update interval of the Status Monitor.

Format

`set monitor update interval` *{count}*

Usage

- *count* specifies the number of seconds between updates of the Status Monitor screen. The default value is 10 seconds. The allowable range of this value is from 5 to 65535 seconds. The command can only be issued to the local system.

This command takes effect after the next update to the Status Monitor display or when the Status Monitor is enabled. This value is saved in non-volatile memory.

7.1.29 Display monitor update interval

Function

This command is used to display the value of the update interval of the Status Monitor.

Format

`display monitor update interval`

Usage

The output of this command is:

```
Monitor Update Interval = 15
```

7.1.30 Add FDB source priority range entry

Function

This command is used to specify the range of source MAC addresses that will have high priority associated with them.

Format

`add FDB source priority range entry` *`{start_address} {end_address}`*

Usage

- *start_address* is the starting source MAC address of the range. The source MAC address is a 48-bit hexadecimal number. The administrator can add up to 64 range entries to the FDB Source Address Priority Range Table. A single range can define up to 65535 MAC addresses. To add a single address to the table, the *start_address* and *end_address* should be set to the same value.

- *end_address* is the ending source MAC address of the range.

This command takes effect immediately. The FDB Source Address Priority Range table is empty by default (i.e., all packets have low priority by default.)

7.1.31 Delete FDB source priority range entry

Function

This command is used to delete a range of source MAC address priority entries from the FDB Source Address Priority Range Table.

Format

`delete fdb source priority range entry` *`{start_address}`*

Usage

- *start_address* is the starting source MAC address of the range.

This command takes effect immediately. The FDB Source Address Priority Range table is empty by default.

7.1.32 Display FDB source priority range table

Function

This command is used to display the contents of the FDB Source Address Priority Range Table.

Format

`display fdb source priority range table`

Usage

A sample output of this command will appear as follows:

```
Begin MAC Address    End MAC Address
01:02:03:04:05:06    01:05:03:04:05:06
02:02:02:02:02:02    02:80:02:02:02:02
End of Table
```

7.1.33 Add FDB destination priority range entry

Function

This command is used to specify the range of destination MAC addresses that will have high priority associated with them.

Format

`add fdb destination priority range entry` *`{start_address} {end_address}`*

Usage

- *start_address* is the starting destination MAC address of the range. The destination MAC address is a 48-bit hexadecimal number. The administrator can add up to 64 range entries to the FDB Destination Address Priority Range Table. A single range can define up to 65535 MAC addresses. To add a single address to the table, the *start_address* and *end_address* should be set to the same value.

- *end_address* is the ending destination MAC address of the range.

This command takes effect immediately. The FDB Destination Address Priority Range table is empty by default (i.e., all packets have low priority by default.)

7.1.34 Delete FDB destination priority range entry

Function

This command is used to delete a range of destination MAC address priority entries from the FDB Destination Address Priority Range Table.

Format

`delete fdb destination priority range entry` *`{start_address}`*

Usage

- *start_address* is the starting destination MAC address of the range.

This command takes effect immediately. The FDB Destination Address Priority Range table is empty by default.

7.1.35 Display FDB destination priority range table

Function

This command is used to display the contents of the FDB Destination Address Priority Range Table.

Format

display fdb destination priority range table

Usage

A sample output of this command will appear as follows:

```
Begin MAC Address    End MAC Address
01:02:03:04:05:06    01:05:03:04:05:06
02:02:02:02:02:02    02:80:02:02:02:02
End of Table
```

7.1.36 Add FDB range entry

Function

This command is used to add an entry into the FDB range table and to add entries within the range to the FDB.

Format

add fdb range entry *{start_address} {end_address}* [**forward** | **discard** | **flood**] *{port_id}*

Usage

Parameters and variables are defined as follows:

- *start_address* is the starting destination MAC address of the range. The destination MAC address is a 48-bit hexadecimal number. The **add fdb range entry** command specifies how the **IDN BRIDGE** handles packets it receives that are intended for a local or remote MAC address. The administrator can add up to 64 range entries to the FDB range table. A single range can define up to 65535 MAC addresses. The FDB range table is preset to empty.

- *end_address* is the ending destination MAC address of the range.

- **forward**, **discard**, and **flood** are the options for the disposition of the packets:

  forward causes the **IDN BRIDGE** to forward packets to the port specified by *port_id*.

  discard causes the **IDN BRIDGE** to discard packets expected to be received on *port_id*.

flood causes the **IDN BRIDGE** to send packets to *port_id*s other than the one on which they were received.

- *port_id* identifies the **IDN BRIDGE** port.

This command takes effect immediately.

7.1.37 Delete FDB range entry

Function

This command is used to delete range entries from the FDB range table.

Format

`delete fdb range entry` *`{start_address}`*

Usage

Parameters and variables are defined as follows:

- *start_address* is the starting destination MAC address of the range.

This command takes effect immediately.

7.1.38 Display FDB range table

Function

This command is used to display the contents of the FDB range table.

Format

`display fdb range table`

Usage

A sample output of this command appears as follows:

```
Begin MAC Address       End MAC Address   DispositionPort
01:02:03:04:05:06       01:05:03:04:05:06 DISCARD    1
02:02:02:02:02:02       02:80:02:02:02:02 FLOOD      1
End of Table
```

7.1.39 Set physical port compression

Function

This is not a new command to the IDN Bridge. If the hardware compression daughter card is detected the compression uses the STAC compression routines. This command defines the default data compression parameters for the specified physical port. The data link or network layer protocol configured for this port

uses these parameters as the defaults for neighbors which have not been defined in the data compression neighbor table.

Format

set physical port compression *{port_id} {status] {max_bfrs}*

Usage

Parameters and variables are defined as follows:

- *port_id* identifies the physical network interface. To see the names and types of physical ports on your router, use the display physical port table command. A physical port name can be changed with the set physical port name command.
- *status*

  on sets the default compression mode to on.

  off sets the default compression mode to off.

  disabled completely disables compression for the physical port. To reenable compression on a physical port that has had compression disabled, enter set physical port compression. The default value is disabled.

- *max_bfrs* specifies the maximum number of history buffers that can be supported for the specified physical port. If the number of active history buffers exceeds *max_bfrs*, newly activated history buffers will operate in packet mode. If a value greater than 1 is specified, and the configured protocol does not support multiple history buffers, only one history buffer will be allocated. Each history buffer uses approximately 12500 bytes of memory. The range of possible values is 1 to 65535. The default for this parameter is 1 for all protocols.

This command requires a reset to take effect.

7.2 Management Information Base

The following section defines the MIB for the Reuters Specific variables. This MIB cleanly passes through common MIB compilers such as mosy and smic.

7.2.1 MIB Organization

```
ACC-SYSTEMS-MIB DEFINITIONS ::= BEGIN

IMPORTS
          enterprises, Counter, Gauge, IpAddress,
          NetworkAddress, TimeTicks
          FROM RFC1155-SMI;
-- Title: ACC SYSTEMS MIB
-- Edition: "Tavarua, November 10, 1999"
-- By: Kevin Stocksdale (kstocks@acc.com)
--
--
--OBJECT DEFINTIONS:  ACC-SYSTEMS PRIVATE MIB
            acc                 OBJECT IDENTIFIER ::= { enterprises 5 }
                accMD           OBJECT IDENTIFIER ::= { acc 2 }
                    accSYS      OBJECT IDENTIFIER ::= { accMD  1 }
-- Reuters Monitoring Group
     accReutersMon   OBJECT IDENTIFIER  ::= { accSYS 1 }
        accReutersMonNum        OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMon 1 }
                        -- Number of physical ports on this unit.
        accReutersMonParmTable  OBJECT-TYPE
                        SYNTAX SEQUENCE OF AccReutersMonParmEntry
                        ACCESS not-accessible
                        STATUS mandatory
                        ::= { accReutersMon 2 }
                        -- A table of parameters of physical ports
                        -- The number of table entries is accReutersMonNum.
            AccReutersMonParmEntry ::= SEQUENCE {
                        accReutersMonPortNo                 INTEGER,
                        accReutersMonHDLCQualityThrs        INTEGER,
                        accReutersMonHDLCInactivityFlag     INTEGER,
                        accReutersMonMLinkLostThrs          INTEGER,
                        accReutersMonEnetDelThrs            INTEGER,
                        accReutersMonEnetTxThrs             INTEGER,
                        accReutersMonEnetIPacketTime        INTEGER,
                        accReutersMonMLinkPackingType       INTEGER,
                        accReutersMonMLinkPackingTMO        INTEGER,
                        accReutersMonMLinkDelayDelta        INTEGER,
                        accReutersMonMLinkState             INTEGER,
                        accReutersMonHDLCQualitySample      INTEGER,
                        accReutersMonHelloPoll              INTEGER

                        }
          accReutersMonParmEntry  OBJECT-TYPE
                        SYNTAX AccReutersMonParmEntry
                        ACCESS not-accessible
                        STATUS mandatory
                        ::= { accReutersMonParmTable 1 }
                        -- Parameters relative to a specific physcial port.
                        -- The instance of an entry is the port number
            accReutersMonPortNo  OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonParmEntry 1 }
                        -- Number of this physical port.
            accReutersMonHDLCQualityThrs  OBJECT-TYPE
                        SYNTAX  INTEGER
```

```
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 2 }
            -- HDLC Quality error threshold
accReutersMonHDLCInactivityFlag  OBJECT-TYPE
            SYNTAX  INTEGER (1..2)
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 3 }
            -- HDLC inactivity monitoring flag
            -- on(1), HDLC inactivity monitoring is active
            -- off(2) HDLC inactivity monitoring is inactive
accReutersMonMLinkLostThrs  OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 4 }
            -- Number of multilink frames lost threshold
accReutersMonEnetDelThrs  OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 5 }
            -- Number of ethernet frames deleted or missed threshold
accReutersMonEnetTxThrs  OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 6 }
            -- Number of consecutive errors when trying to
            -- transmit a packet threshold
accReutersMonEnetIPacketTime  OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 7 }
            -- Number of seconds in which an ethernet packet
            -- should be received
accReutersMonMLinkPackingType OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 8 }
            -- Frame Packing type
accReutersMonMLinkPackingTMO OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 9 }
            -- Frame Packing Time-out
accReutersMonMLinkDelayDelta OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 10 }
            -- WAN Link Override Delay Delta (msec)
accReutersMonMLinkState      OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-only
            STATUS  mandatory
            ::= { accReutersMonParmEntry 11 }
            -- WAN Link State
            -- down(0),   WAN Link is down
            -- setup(1),  WAN Link is setting up
            -- up(2),     WAN Link is operational
            -- stdby(3)  WAN Link is standby
accReutersMonHDLCQualitySample  OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            ::= { accReutersMonParmEntry 12 }
             --HDLC Quality sample count
accReutersMonHelloPoll  OBJECT-TYPE
```

```
                SYNTAX  INTEGER
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonParmEntry 13 }
                -- LAN/WAN Hello Poll interval

accReutersMonAlarmTable  OBJECT-TYPE
                SYNTAX SEQUENCE OF AccReutersMonAlarmEntry
                ACCESS not-accessible
                STATUS mandatory
                ::= { accReutersMon 3 }
                -- A list of per-port monitoring alarms
        AccReutersMonAlarmEntry ::= SEQUENCE {
                accReutersMonAlarmIndex            INTEGER,
                accReutersMonHDLCQualityAlarm      INTEGER,
                accReutersMonHDLCInactivityAlarm   INTEGER,
                accReutersMonMLinkLostAlarm        INTEGER,
                accReutersMonEnetDelAlarm          INTEGER,
                accReutersMonEnetTxAlarm           INTEGER,
                accReutersMonEnetIPacketTimeAlarm  INTEGER
                }
  accReutersMonAlarmEntry  OBJECT-TYPE
                SYNTAX AccReutersMonAlarmEntry
                ACCESS not-accessible
                STATUS mandatory
                ::= { accReutersMonAlarmTable 1 }
                -- Monitoring alarm flags
    accReutersMonAlarmIndex     OBJECT-TYPE
                SYNTAX  INTEGER
                ACCESS  read-only
                STATUS  mandatory
                ::= { accReute rsMonAlarmEntry 1 }
                 -- Port number of this entry
    accReutersMonHDLCQualityAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..3)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 2 }
                -- Status of HDLC Quality monitoring alarm
                -- on(1),  alarm is active
                -- off(2)  alarm is inactive
                -- ansi(3)  alarm is active with ansi screen
    accReutersMonHDLCInactivityAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 3 }
                -- Status of HDLC Inactivity alarm
                -- on(1),  Alarm is active
                -- off(2)  Alarm is inactive
    accReutersMonMLinkLostAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 4 }
                -- Status of Multilink lost frames alarm
                -- on(1),  Alarm is active
                -- off(2)  Alarm is inactive
    accReutersMonEnetDelAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 5 }
                -- Status of ethernet frames lost or deleted alarm
                -- on(1), alarm is active
                -- off(2) alarm is inactive
    accReutersMonEnetTxAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 6 }
                -- Status of ethernet consecutive
                -- transmission error alarm
```

```
                -- on(1),  alarm is active
                -- off(2)  alarm is inactive
    accReutersMonEnetIPacketTimeAlarm  OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-write
                STATUS  mandatory
                ::= { accReutersMonAlarmEntry 7 }
                -- Status of inter-packet time for
                -- received ethernet packets alarm
                -- on(1),  alarm is active
                -- off(2)  alarm is inactive
accReutersMonAlarm      OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-only
                STATUS  mandatory
                ::= { accReutersMon 4 }
                -- Alarm state of the system
                -- on(1),  at least one alarm is active
                -- off(2)  all alarms are inactive
accReutersMonAll        OBJECT-TYPE
                SYNTAX  INTEGER (1..4)
                ACCESS  read-only
                STATUS  mandatory
                ::= { accReutersMon 5 }
                -- Alarm state and live standby state of system
                -- livenoalarm(1),    live w/ no alarms
                -- livealarm(2),      live w/ alarm
                -- standbynoalarm(3), standby w/ no alarms
                -- standbyalarm(4)    standby w/ alarm
accReutersMonAllString  OBJECT-TYPE
                SYNTAX  OCTET STRING
                ACCESS  read-only
                STATUS  mandatory
                ::= { accReutersMon 6 }
                -- Alarm state and live standby state of system
                -- Live W/ No Alarm
                -- Live W/ Alarm
                -- Standby W/ No Alarm
                -- Standby W/ Alarm
accReutersMonLiveStandby        OBJECT-TYPE
                SYNTAX  INTEGER (1..2)
                ACCESS  read-only
                STATUS  mandatory
                ::= { accReutersMon 7 }
                -- Live/standby indicator of the system
                -- on(0),  unit is live system
                -- off(1)  unit is standby system
accReutersMonLAPBUTable  OBJECT-TYPE
                SYNTAX SEQUENCE OF AccReutersMonLAPBUEntry
                ACCESS not-accessible
                STATUS mandatory
                ::= { accReutersMon 8 }
                -- A list of per-port LAPB utilization parameters
        AccReutersMonLAPBUEntry ::= SEQUENCE {
                accReutersMonLAPBUIndex            INTEGER,
                accReutersMonLAPBUCurrent          INTEGER,
                accReutersMonLAPBUHigh             INTEGER,
                accReutersMonLAPBUSpeed            INTEGER,
                accReutersMonLAPBUHighWater        INTEGER,
                accReutersMonLAPBTransitDelay      INTEGER,
                accReutersMonLAPBTransitMulti      INTEGER,
                accReutersMonLAPB2lq               INTEGER,
                accReutersMonAvgPktSz              INTEGER,
                accReutersMonLAPBTestTMO           INTEGER,
                accReutersMonLAPBT1Variance        INTEGER
                }
  accReutersMonLAPBUEntry  OBJECT-TYPE
                SYNTAX AccReutersMonLAPBUEntry
                ACCESS not-accessible
                STATUS mandatory
                ::= { accReutersMonLAPBUTable 1 }
                -- LAPB utilization parameters.
                -- The instance of an entry is the
```

```
                        -- accReutersMonLAPBUIndex, which may range from
                        -- zero to number of phsical ports.
                accReutersMonLAPBUIndex     OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 1 }
                        -- Port number of this entry
                accReutersMonLAPBUCurrent   OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 2 }
                        -- Current LAPB utilization value
                accReutersMonLAPBUHigh      OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 3 }
                        -- Highest LAPB utilization value since last reset
                accReutersMonLAPBUSpeed     OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 4 }
                        -- Configured LAPB port speed
                accReutersMonLAPBUHighWater OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 5 }
                        -- LAPB utilization percentage high water threshold
                accReutersMonLAPBTransitDelay OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 6 }
                        -- LAPB Transit Delay in milli-seconds
                accReutersMon LAPBTransitMulti OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 7 }
                        -- LAPB Transit Delay Multiplier for T1 Timer
                accReutersMonLAPB2lq OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 8 }
                        -- LAPB Second Level Queue Threshold
                accReutersMonAvgPktSz OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 9 }
                        -- Average Packet Size
                accReutersMonLAPBTestTMO OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 10 }
                       -- LAPB Test Msg TMO
                accReutersMonLAPBT1Variance OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersMonLAPBUEntry 11 }
                        -- LAPB T1 Timer Variance

        -- Bridge FDB Range Group
              accFDBRange OBJECT IDENTIFIER ::= { accSYS 2 }
                accFDBRangeTable   OBJECT-TYPE
                        SYNTAX SEQUENCE OF AccFDBRangeEntry
```

```
                        ACCESS not-accessible
                        STATUS mandatory
                         ::= { accFDBRange 1 }
                        -- A list of Forwarding Database range entries.
                        -- Entries may be added/deleted either through
                        -- explicit management action.
                  AccFDBRangeEntry  ::= SEQUENCE {
                        accFDBRangeEntBegMacAddr      OCTET STRING,
                        accFDBRangeEntEndMacAddr      OCTET STRING,
                        accFDBRangeEntDisp            INTEGER,
                        accFDBRangeEntPort            INTEGER
                        }
            accFDBRangeEntry OBJECT-TYPE
                        SYNTAX AccFDBRangeEntry
                        ACCESS not-accessible
                        STATUS mandatory
                        ::= { accFDBRangeTable 1 }
                        -- A Forwarding Database range entry which defines
                        -- Bridge action to be taken based upon the
                        -- destination MAC Address of received frames.
                        -- The instance of a range entry is the
                        -- accFDBRangeEntBegMacAddr.
              accFDBRangeEntBegMacAddr  OBJECT-TYPE
                        SYNTAX  OCTET STRING
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accFDBRangeEntry 1 }
                        -- Beginning MAC Address of the Forwarding
                        -- Database Range Entry.  Object required in
                        -- entry-adding "SET" PDU.  Set this object to
                        -- NULL to delete range entry.
              accFDBRangeEntEndMacAddr  OBJECT-TYPE
                        SYNTAX  OCTET STRING
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accFDBRangeEntry 2 }
                        -- Ending MAC Address of the Forwarding Database
                        -- Range Entry.
              accFDBRangeEntDisp     OBJECT-TYPE
                        SYNTAX  INTEGER (0..2)
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accFDBRangeEntry 3 }
                        -- Disposition of received Frame. Note that a
                        -- Frame is never sent back out on the Port on
                        -- which it was received.  Object required in
                        -- entry-adding "SET" PDU.
                        -- forward(0),
                        -- flood(1),
                        -- discard(2)
              accFDBRangeEntPort     OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accFDBRangeEntry 4 }
                        -- Port to which frame is to be relayed for
                        -- frame to be forwarded.  Has no meaning for
                        -- other dispositions.
                        -- Object required in entry-adding "SET" PDU.
      -- Bridge Priority Range Group
            accPriorityRange OBJECT IDENTIFIER ::= { accSYS 3 }
              accSourcePriorityRangeTable   OBJECT-TYPE
                        SYNTAX SEQUENCE OF AccSourcePriorityRangeEntry
                        ACCESS not-accessible
                        STATUS mandatory
                        ::= { accPriorityRange 1 }
                        -- A list of Source Address Priority Range entries.
                    AccSourcePriorityRangeEntry  ::= SEQUENCE {
                        accSourcePriorityRangeEntBegMacAddr      OCTET STRING,
                        accSourcePriorityRangeEntEndMacAddr      OCTET STRING
                        }
                  accSourcePriorityRangeEntry OBJECT-TYPE
                        SYNTAX AccSourcePriorityRangeEntry
```

```
                    ACCESS not-accessible
                    STATUS mandatory
                    ::= { accSourcePriorityRangeTable 1 }
                    -- A Source Address Priority Range Entry.
                    -- The instance of an entry
                    -- is the accSourcePriorityRangeEntBegMacAddr.
      accSourcePriorityRangeEntBegMacAddr  OBJECT-TYPE
                    SYNTAX  OCTET STRING
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accSourcePriorityRangeEntry 1 }
                    -- Beginning MAC Address of the range.
                    -- Object required in entry-adding "SET" PDU.
                    -- Set this object to NULL to delete entry.
      accSourcePriorityRangeEntEndMacAddr  OBJECT-TYPE
                    SYNTAX  OCTET STRING
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accSourcePriorityRangeEntry 2 }
                    -- Ending MAC Address of the range.
  accDestinationPriorityRangeTable   OBJECT-TYPE
                    SYNTAX SEQUENCE OF AccDestinationPriorityRangeEntry
                    ACCESS not-accessible
                    STATUS mandatory
                    ::= { accPriorityRange 2 }
                    -- A list of Destination Address Priority
                    -- Range entries.
          AccDestinationPriorityRangeEntry  ::= SEQUENCE {
                    accDestinationPriorityRangeEntBegMacAddr      OCTET STRING,
                    accDestinationPriorityRangeEntEndMacAddr      OCTET STRING
                    }
    accDestinationPriorityRangeEntry OBJECT-TYPE
                    SYNTAX AccDestinationPriorityRangeEntry
                    ACCESS not-accessible
                    STATUS mandatory
                    ::= { accDestinationPriorityRangeTable 1 }
                    -- A Destination Address Priority Range Entry.
                    -- The instance of an entry
                    -- is the accDestinationPriorityRangeEntBegMacAddr.
      accDestinationPriorityRangeEntBegMacAddr  OBJECT-TYPE
                    SYNTAX  OCTET STRING
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accDestinationPriorityRangeEntry 1 }
                    -- Beginning MAC Address of the range.
                    -- Object required in entry-adding "SET" PDU.
                    -- Set this object to NULL to delete entry.
      accDestinationPriorityRangeEntEndMacAddr  OBJECT-TYPE
                    SYNTAX  OCTET STRING
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accDestinationPriorityRangeEntry 2 }
                    -- Ending MAC Address of the range.
      accReutersMonBufferHighWM      OBJECT-TYPE
                    SYNTAX  INTEGER
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accPriorityRange 5 }
                    -- Buffer monitoring high water mark
      accReutersMonBufferLowWM      OBJECT-TYPE
                    SYNTAX  INTEGER
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accPriorityRange 6 }
                    -- Buffer monitoring low water mark
      accReutersMonBufferHighest        OBJECT-TYPE
                    SYNTAX  INTEGER
                    ACCESS  read-write
                    STATUS  mandatory
                    ::= { accPriorityRange 7 }
                    -- Buffer monitoring highest level
      accReutersMonBufferCurrent        OBJECT-TYPE
                    SYNTAX  INTEGER
```

```
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 8 }
                        -- Buffer monitoring current value
                accReutersCpuAvgLoad              OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 9 }
                        -- CPU Load Average current value
                accReutersCpuMaxLoad              OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 10 }
                        -- CPU Load Average highest value
                accReutersCpuThrLoad              OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 11 }
                        -- CPU Load Average alarm threshold
                accReutersBrTyps                  OBJECT-TYPE
                        SYNTAX  INTEGER (0..2)
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 12 }
                        -- Bridge Type
                        -- none(0),
                        -- north(1),
                        -- south(2),
                accReutersBrStat                  OBJECT-TYPE
                        SYNTAX  INTEGER (1..2)
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accPriorityRange 13 }
                        -- Bridge Standby
                        -- Write nop(1),   do nothing
                        -- Write standby(2), force bridge to standby
                accReutersLowPriorityDiscards       OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        ::= { accPriorityRange 14 }
                        -- number of low priority frames which were discarded
                        -- due to exceeding the memory utilization threshold
-- Reuters Clear Group
     accReutersClear   OBJECT IDENTIFIER  ::= { accSYS 4 }
        accReutersClearAlarm      OBJECT-TYPE
                        SYNTAX  INTEGER (1..2)
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersClear 1 }
                        -- Clear Alarm state of the system
                        -- Write nop(1),   do nothing
                        -- Write reset(2), clear all alarms
        accReutersClearStats      OBJECT-TYPE
                        SYNTAX  INTEGER (1..2)
                        ACCESS  read-write
                        STATUS  mandatory
                        ::= { accReutersClear 2 }
                        -- Clear Miscellaneous Monitor Stats
                        -- Write nop(1),   do nothing
                        -- Write reset(2), clear all alarms
END
```

7.2.2 SNMP Management Traps

See section 4.1.2.2

A.1 CLISPEC/CLIGEN Input

```
<Product>
   ProductId  { REUTERS_CMDS }
   ProductOID { 43.6.1.4.1.5.1.1.1.1.15.5.16.11 }
   ProductDesc { Reuters 11.0 Special }
   ModeFlags { DYNPRT, IFGRP }   // only possible values at this time
   ExistCmds  {
               /*
               * Following is list of "existing" commands
               * used by this product
               */
              AD_AC_EN,      //ADD ACCESS ENTRY
              AD_AR_EN,      //ADD ARP ENTRY
              AD_BO_SE_EN,   //ADD BOOTP SERVER ENTRY
              AD_BR_DI_OR_EN,        //ADD BRIDGE DIAL ORIGINATE ENTRY
              AD_BR_FI_EN_MSK,       //ADD BRIDGE FILTER ENTRY
              AD_BR_PA_FI_EN,        //ADD BRIDGE PATTERN FILTER ENTRY
              AD_BR_PO_EN_DI,        //ADD BRIDGE PORT ENTRY DIAL
              AD_BR_PO_EN_ET,        //ADD BRIDGE PORT ENTRY ETHERNET
              AD_BR_PO_EN_FR,        //ADD BRIDGE PORT ENTRY FR
              AD_BR_PO_EN_LA,        //ADD BRIDGE PORT ENTRY LAPB
              AD_BR_PO_EN_MU,        //ADD BRIDGE PORT ENTRY MULTILINK
              AD_BR_PO_EN_PP,        //ADD BRIDGE PORT ENTRY PPP
              AD_CO_FF_EN,   //ADD COMPRESSION FFR ENTRY
              AD_DI_BA_EN,   //ADD DIAL BACKUP ENTRY
              AD_DI_PO_CA_AD_CHAP,   //ADD DIAL PORT CALL ADDRESS
              AD_DI_PO_EN,   //ADD DIAL PORT ENTRY
              AD_DI_PO_PH_PO,        //ADD DIAL PORT PHYSICAL PORT
              AD_DO_HO_EN,   //ADD DOWNLOAD HOST ENTRY
              AD_FD_EN,      //ADD FDB ENTRY
              AD_FR_SW_EN,   //ADD FRAME_RELAY SWITCH ENTRY
              AD_IP_AL_EN,   //ADD IP ALIAS ENTRY
              AD_IP_AS_AD_RA,        //ADD IP ASSIGNED ADDRESS RANGE
              AD_IP_CI_RO_EN,        //ADD IP ROUTE ENTRY
              AD_IP_DI_FI,   //ADD IP DISCARD FILTER ENTRY
              AD_IP_DI_OR_EN,        //ADD IP DIAL ORIGINATE ENTRY
              AD_IP_FF_EN,   //ADD IP FFR_NEIGHBOR ENTRY
              AD_IP_MA_EN_0, //ADD IP MAP ENTRY
              AD_IP_MA_NAV,  //ADD IP MAP SERVER ENTRY
              AD_IP_NE_EN_0, //ADD IP NETWORK ENTRY
              AD_MU_GR_EN,   //ADD MULTILINK GROUP ENTRY
              AD_MU_GR_PH_PO,        //ADD MULTILINK GROUP PHYSICAL PORT
              AD_PR_EL,      //ADD PROFILE ELEMENTS
              AD_PR_EN,      //ADD PROFILE ENTRY
              AD_PR_GL_AS,   //ADD PROFILE GLOBAL ASSIGNMENT
              AD_PR_IP_FI_EN,        //ADD PROFILE IP FILTER ENTRY
              AD_PR_PO_AS,   //ADD PROFILE PORT ASSIGNMENT
              AD_PR_PR_EN,   //ADD PROTOCOL PRIORITY ENTRY
              AD_TR_EN,      //ADD TRAP ENTRY
              AD_UDP_HE_EN,  //ADD UDP HELPER ENTRY
              DE_AC_EN,      //DELETE ACCESS ENTRY
              DE_AR_EN,      //DELETE ARP ENTRY
              DE_BO_SE_EN,   //DELETE BOOTP SERVER ENTRY
              DE_BR_DI_OR_EN,        //DELETE BRIDGE DIAL ORIGINATE ENTRY
              DE_BR_FI_EN_MSK,       //DELETE BRIDGE FILTER ENTRY
              DE_BR_PA_FI_EN,        //DELETE BRIDGE PATTERN FILTER ENTRY
              DE_BR_PO_EN,   //DELETE BRIDGE PORT ENTRY
              DE_CO_FF_EN,   //DELETE COMPRESSION FFR ENTRY
              DE_DI_BA_EN,   //DELETE DIAL BACKUP ENTRY
              DE_DI_PO_CA_AD,        //DELETE DIAL PORT CALL ADDRESS
              DE_DI_PO_EN,   //DELETE DIAL PORT ENTRY
              DE_DI_PO_PH_PO,        //DELETE DIAL PORT PHYSICAL PORT
              DE_DO_HO_EN,   //DELETE DOWNLOAD HOST ENTRY
              DE_FD_EN,      //DELETE FDB ENTRY
              DE_FF_CI_EN,   //DELETE FFR CIRCUIT ENTRY
              DE_FI_EN,      //DELETE FILE ENTRY
              DE_FR_SW_EN,   //DELETE FRAME_RELAY SWITCH ENTRY
              DE_IP_AL_EN,   //DELETE IP ALIAS ENTRY
              DE_IP_AS_AD_RA,        //DELETE IP ASSIGNED ADDRESS RANGE
```

```
DE_IP_CI_RO_EN,        //DELETE IP ROUTE ENTRY
DE_IP_DI_FI,   //DELETE IP DISCARD FILTER ENTRY
DE_IP_DI_OR_EN,        //DELETE IP DIAL ORIGINATE ENTRY
DE_IP_FF_EN,   //DELETE IP FFR_NEIGHBOR ENTRY
DE_IP_MA_EN,   //DELETE IP MAP ENTRY
DE_IP_MA_NAV,  //DELETE IP MAP SERVER ENTRY
DE_IP_NE_EN_0, //DELETE IP NETWORK ENTRY
DE_MU_GR_EN,   //DELETE MULTILINK GROUP ENTRY
DE_MU_GR_PH_PO,        //DELETE MULTILINK GROUP PHYSICAL PORT
DE_PR_EL,      //DELETE PROFILE ELEMENTS
DE_PR_EN,      //DELETE PROFILE ENTRY
DE_PR_GL_AS,   //DELETE PROFILE GLOBAL ASSIGNMENT
DE_PR_IP_FI_EN,        //DELETE PROFILE IP FILTER ENTRY
DE_PR_PO_AS,   //DELETE PROFILE PORT ASSIGNMENT
DE_PR_PR_EN,   //DELETE PROTOCOL PRIORITY ENTRY
DE_TR_EN,      //DELETE TRAP ENTRY
DE_UDP_HE_EN,  //DELETE UDP HELPER ENTRY
DI_AC_TA,      //DISPLAY ACCESS TABLE
DI_AR_EN_1,    //DISPLAY ARP ENTRY
DI_AR_PA,      //DISPLAY ARP PARAMETERS
DI_AR_ST,      //DISPLAY ARP STATISTICS
DI_AR_TA_5,    //DISPLAY ARP TABLE
DI_AS_PA_TA,   //DISPLAY ASYNC PARAMETER TABLE
DI_AS_ST_TA,   //DISPLAY ASYNC STATISTICS TABLE
DI_BO_SE_TA,   //DISPLAY BOOTP SERVER TABLE
DI_BO_ST_81,   //DISPLAY BOOTP STATUS
DI_BR_CO_MO,   //DISPLAY BRIDGE COMPRESSION MODE
DI_BR_DI_OR_EN,        //DISPLAY BRIDGE DIAL ORIGINATE ENTRY
DI_BR_DI_OR_TA,        //DISPLAY BRIDGE DIAL ORIGINATE TABLE
DI_BR_FI_EN_MSK,       //DISPLAY BRIDGE FILTER ENTRY
DI_BR_FI_PA,   //DISPLAY BRIDGE FILTER PARAMETERS
DI_BR_FI_ST,   //DISPLAY BRIDGE FILTER STATISTICS
DI_BR_FI_TA_MSK,       //DISPLAY BRIDGE FILTER TABLE
DI_BR_LE_MO,   //DISPLAY BRIDGE LEARNING MODE
DI_BR_PA_4,    //DISPLAY BRIDGE PARAMETERS
DI_BR_PA_FI_EN,        //DISPLAY BRIDGE PATTERN FILTER ENTRY
DI_BR_PA_FI_TA,        //DISPLAY BRIDGE PATTERN FILTER TABLE
DI_BR_PO_FR_TA,        //DISPLAY BRIDGE PORT FR TABLE
DI_BR_PO_TA_2, //DISPLAY BRIDGE PORT TABLE
DI_COM_FF_EN,  //DISPLAY COMPRESSION FFR ENTRY
DI_COM_FF_ST_EN,       //DISPLAY COMPRESSION FFR STATISTICS ENTRY
DI_COM_FF_ST_TA,       //DISPLAY COMPRESSION FFR STATISTICS TABLE
DI_COM_FF_TA,  //DISPLAY COMPRESSION FFR TABLE
DI_CONS_SP,    //DISPLAY CONSOLE SPEED
DI_DI_BA_EN,   //DISPLAY DIAL BACKUP ENTRY
DI_DI_BA_TA,   //DISPLAY DIAL BACKUP TABLE
DI_DI_COM_EN,  //DISPLAY DIAL PORT COMPRESSION ENTRY
DI_DI_COM_TA,  //DISPLAY DIAL PORT COMPRESSION TABLE
DI_DI_PO_EN,   //DISPLAY DIAL PORT ENTRY
DI_DI_PO_RA,   //DISPLAY DIAL PORT RANGE
DI_DI_PO_TA,   //DISPLAY DIAL PORT TABLE
DI_DI_PO_ST_EN_90,     //DISPLAY DIAL PORT STATUS ENTRY
DI_DI_PO_ST_TA_90,     //DISPLAY DIAL PORT STATUS TABLE
DI_DI_US_TA,   //DISPLAY DIAL PORT CONNECTION TABLE
DI_DO_HO_TA,   //DISPLAY DOWNLOAD HOST TABLE
DI_EN_ST_TA_0, //DISPLAY ENET STATISTICS TABLE
DI_ET_ST_TA_0, //DISPLAY ETHERNET STATISTICS TABLE
DI_FDB_EN,     //DISPLAY FDB ENTRY
DI_FDB_PA,     //DISPLAY FDB PARAMETERS
DI_FDB_TA,     //DISPLAY FDB TABLE
DI_FF_CI_EN_2, //DISPLAY FFR CIRCUIT ENTRY
DI_FF_CI_ST_TA,        //DISPLAY FFR CIRCUIT STATISTICS TABLE
DI_FF_CI_TA_4, //DISPLAY FFR CIRCUIT TABLE
DI_FF_ER_TA,   //DISPLAY FFR ERROR TABLE
DI_FF_PA_TA_2, //DISPLAY FFR PARAMETER TABLE
DI_FF_ST_TA,   //DISPLAY FFR STATISTICS TABLE
DI_FI_CO,      //DISPLAY FILE CONFIG
DI_FI_EN,      //DISPLAY FILE ENTRY
DI_FI_SU,      //DISPLAY FILE SUMMARY
DI_FI_SY_EN,   //DISPLAY FILE SYSTEM ENTRY
DI_FI_SY_TA,   //DISPLAY FILE SYSTEM TABLE
DI_FL_TA,      //DISPLAY FLASH TABLE
DI_FR_SW_EN,   //DISPLAY FRAME_RELAY SWITCH ENTRY
```

```
                DI_FR_SW_TA,   //DISPLAY FRAME_RELAY SWITCH TABLE
                DI_IC_ST,      //DISPLAY ICMP STATISTICS
                DI_IP_AD_NE_TA,        //DISPLAY IP ADDRESS NEGOTIATION TABLE
                DI_IP_AL_TA,   //DISPLAY IP ALIAS TABLE
                DI_IP_AS_AD_PO,        //DISPLAY IP ASSIGNED ADDRESS POOL
                DI_IP_AS_AD_ST,        //DISPLAY IP ASSIGNED ADDRESS STATISTICS
                DI_IP_CI_RO_AP_TA,     //DISPLAY IP ROUTE APPLICATION TABLE
                DI_IP_CI_RO_EN,        //DISPLAY IP ROUTE ENTRY
                DI_IP_CI_RO_TA,        //DISPLAY IP ROUTE TABLE
                DI_IP_DI_FI_TA,        //DISPLAY IP DISCARD FILTER TABLE
                DI_IP_DI_OR_EN,        //DISPLAY IP DIAL ORIGINATE ENTRY
                DI_IP_DI_OR_TA,        //DISPLAY IP DIAL ORIGINATE TABLE
                DI_IP_FF_EN,   //DISPLAY IP FFR_NEIGHBOR ENTRY
                DI_IP_FF_TA,   //DISPLAY IP FFR_NEIGHBOR TABLE
                DI_IP_MA_EN_0, //DISPLAY IP MAP ENTRY
                DI_IP_MA_ST_TA,        //DISPLAY IP MAP STATISTICS TABLE
                DI_IP_MA_TA,   //DISPLAY IP MAP TABLE
                DI_IP_ME_ST_TA,        //DISPLAY IP MEMORY STATISTICS TABLE
                DI_IP_NAV_EN,  //DISPLAY IP MAP SERVER ENTRY
                DI_IP_NAV_TA,  //DISPLAY IP MAP SERVER TABLE
                DI_IP_NE_EN_0, //DISPLAY IP NETWORK ENTRY
                DI_IP_NE_SE_2, //DISPLAY IP NETWORK SECURITY
                DI_IP_NE_TA,   //DISPLAY IP NETWORK TABLE
                DI_IP_PA,      //DISPLAY IP PARAMETERS
                DI_IP_RT_CA_TA,        //DISPLAY IP ROUTE CACHE TABLE
                DI_IP_RT_IN,   //DISPLAY IP ROUTE STATISTICS
                DI_IP_SO_RO,   //DISPLAY IP SOURCE ROUTING
                DI_IP_ST,      //DISPLAY IP STATISTICS
                DI_IP_UN_DE_SO_AD,     //DISPLAY IP UNNUMBERED INTERFACE DEFAULT SOURCE
ADDRESS
                DI_IP_UN_IN_TA,        //DISPLAY IP UNNUMBERED INTERFACE TABLE
                DI_LA_FC_TA,   //DISPLAY LAPB FCS TABLE
                DI_LA_ST_EN,   //DISPLAY LAPB STATISTICS ENTRY
                DI_LA_ST_TA,   //DISPLAY LAPB STATISTICS TABLE
                DI_LA_TA,      //DISPLAY LAPB TABLE
                DI_ME_DB_IN_TA,        //DISPLAY MEMORY DBLOCK INCREMENT TABLE
                DI_ME_OP_PA,   //DISPLAY MEMORY OPTIMIZER PARAMETERS
                DI_ME_OP_TA,   //DISPLAY MEMORY OPTIMIZER TABLE
                DI_ME_PO_TA,   //DISPLAY MEMORY POOL TABLE
                DI_ME_ST_TA_90,        //DISPLAY MEMORY STATISTICS TABLE
                DI_MU_GR_CO,   //DISPLAY MULTILINK GROUP COUNT
                DI_MU_GR_CO_EN,        //DISPLAY MULTILINK GROUP COMPRESSION ENTRY
                DI_MU_GR_CO_ST_EN,     //DISPLAY MULTILINK GROUP COMPRESSION STATISTICS
ENTRY
                DI_MU_GR_CO_ST_TA,     //DISPLAY MULTILINK GROUP COMPRESSION STATISTICS
TABLE
                DI_MU_GR_CO_TA,        //DISPLAY MULTILINK GROUP COMPRESSION TABLE
                DI_MU_GR_ML_EN,        //DISPLAY MULTILINK GROUP STANDARD PROTOCOL ENTRY
                DI_MU_GR_ML_TA,        //DISPLAY MULTILINK GROUP STANDARD PROTOCOL TABLE
                DI_MU_GR_PA_EN_ENH2,   //DISPLAY MULTILINK GROUP PARAMETER ENTRY
                DI_MU_GR_PA_TA_ENH2,   //DISPLAY MULTILINK GROUP PARAMETER TABLE
                DI_MU_GR_ST_EN,        //DISPLAY MULTILINK GROUP STATISTICS ENTRY
                DI_MU_GR_ST_TA,        //DISPLAY MULTILINK GROUP STATISTICS TABLE
                DI_MU_GR_US_ST_EN,     //DISPLAY MULTILINK GROUP USAGE STATISTICS ENTRY
                DI_MU_GR_US_ST_TA,     //DISPLAY MULTILINK GROUP USAGE STATISTICS TABLE
                DI_NV_ST,      //DISPLAY NVM STATUS
                DI_PH_PO_CO_EN_NOREV,  //DISPLAY PHYSICAL PORT COMPRESSION ENTRY
                DI_PH_PO_CO_ST_EN_CCP, //DISPLAY PHYSICAL PORT COMPRESSION STATISTICS ENTRY
                DI_PH_PO_CO_ST_TA_CCP, //DISPLAY PHYSICAL PORT COMPRESSION STATISTICS TABLE
                DI_PH_PO_CO_TA_NOREV,  //DISPLAY PHYSICAL PORT COMPRESSION TABLE
                DI_PH_PO_E1_TA,        //DISPLAY PHYSICAL PORT E1 TABLE
                DI_PH_PO_EN,   //DISPLAY PHYSICAL PORT ENTRY
                DI_PH_PO_NA_TA_80,     //DISPLAY PHYSICAL PORT NAME TABLE
                DI_PH_PO_RE_TA,        //DISPLAY PHYSICAL PORT RESYNC TABLE
                DI_PH_PO_ST_EN,        //DISPLAY PHYSICAL PORT STATISTICS ENTRY
                DI_PH_PO_ST_TA_1,      //DISPLAY PHYSICAL PORT STATISTICS TABLE
                DI_PH_PO_TA_8, //DISPLAY PHYSICAL PORT TABLE
                DI_PI_PA,      //DISPLAY PING PARAMETERS
                DI_PP_AU_PA_EN,        //DISPLAY PPP AUTHENTICATION PARAMETER ENTRY
                DI_PP_AU_PA_TA,        //DISPLAY PPP AUTHENTICATION PARAMETER TABLE
                DI_PP_AU_ST_EN,        //DISPLAY PPP AUTHENTICATION STATISTICS ENTRY
                DI_PP_AU_ST_TA,        //DISPLAY PPP AUTHENTICATION STATISTICS TABLE
                DI_PP_COM_EN,  //DISPLAY PPP COMPRESSION ENTRY
```

```
DI_PP_COM_ST_EN,       //DISPLAY PPP COMPRESSION STATISTICS ENTRY
DI_PP_COM_ST_TA,       //DISPLAY PPP COMPRESSION STATISTICS TABLE
DI_PP_COM_TA,  //DISPLAY PPP COMPRESSION TABLE
DI_PP_MO_TA,   //DISPLAY PPP MONITORING TABLE
DI_PP_PA_TA_90,        //DISPLAY PPP PARAMETER TABLE
DI_PP_PR_TA,   //DISPLAY PPP PROTOCOL TABLE
DI_PP_ST_TA,   //DISPLAY PPP STATISTICS TABLE
DI_PR_EN,      //DISPLAY PROFILE ENTRY
DI_PR_IP_AP_EN,        //DISPLAY PROFILE IP APPLICATION ENTRY
DI_PR_IP_AP_TA,        //DISPLAY PROFILE IP APPLICATION TABLE
DI_PR_IP_FI_EN9_7,     //DISPLAY PROFILE IP FILTER ENTRY
DI_PR_IP_FI_TA9_7,     //DISPLAY PROFILE IP FILTER TABLE
DI_PR_PA,      //DISPLAY PROFILE PARAMETERS
DI_PR_PO_EN,   //DISPLAY PROFILE PORT ENTRY
DI_PR_PO_TA,   //DISPLAY PROFILE PORT TABLE
DI_PR_PR_DE,   //DISPLAY PROTOCOL PRIORITY DEFAULT
DI_PR_PR_TA,   //DISPLAY PROTOCOL PRIORITY TABLE
DI_PR_TA,      //DISPLAY PROFILE TABLE
DI_SN_AG_ST,   //DISPLAY SNMP AGENT STATISTICS
DI_SN_CL_ST,   //DISPLAY SNMP CLIENT STATISTICS
DI_SN_GE_ST,   //DISPLAY SNMP GENERAL STATISTICS
DI_STA_DI,     //DISPLAY STATISTICS DISPLAY
DI_STP_PA,     //DISPLAY STP PARAMETERS
DI_STP_PO_ST_TA,       //DISPLAY STP PORT STATISTICS TABLE
DI_STP_PO_TA,  //DISPLAY STP PORT TABLE
DI_STP_ST,     //DISPLAY STP STATISTICS
DI_SY_IN,      //DISPLAY SYSTEM INFORMATION
DI_SY_MA_AD,   //DISPLAY SYSTEM MAC ADDRESS
DI_SY_OP_PA,   //DISPLAY SYSTEM OPTIMIZER PARAMETERS
DI_SY_OP_TA,   //DISPLAY SYSTEM OPTIMIZER TABLE
DI_SY_SE_TA,   //DISPLAY SYSTEM SERVICE TABLE
DI_TCP_CO_TA,  //DISPLAY TCP CONNECTION TABLE
DI_TCP_ST,     //DISPLAY TCP STATISTICS
DI_TF_ST,      //DISPLAY TFTP STATUS
DI_TI_80,      //DISPLAY TIME
DI_TRAP_TA,    //DISPLAY TRAP TABLE
DI_TR_LO_PA,   //DISPLAY TRAP LOG PARAMETERS
DI_TR_LO_SU,   //DISPLAY TRAP LOG SUMMARY
DI_TR_LO_TA,   //DISPLAY TRAP LOG TABLE
DI_TR_ST,      //DISPLAY TRACEROUTE STATUS
DI_UDP_HE_PA,  //DISPLAY UDP HELPER PARAMETERS
DI_UDP_HE_STAT_TA,     //D ISPLAY UDP HELPER STATISTICS TABLE
DI_UDP_HE_TA,  //DISPLAY UDP HELPER TABLE
DI_UD_ST,      //DISPLAY UDP STATISTICS
DI_US_TA,      //DISPLAY USER TABLE
DI_V2_ST_TA,   //DISPLAY V25 STATISTICS TABLE
DO,     //DOWNLOAD
LOGIN,  //LOGIN
LOGO,   //LOGOUT
PA,     //PASSWORD
PI,     //PING
REL_FS, //RELOAD
RES,    //RESET
SC,     //SCRIPTLOAD
SE_AR_MO,      //SET ARP MODE
SE_AR_TI,      //SET ARP TIMEOUT
SE_AS_FR_PA,   //SET ASYNC FRAMING PARAMETERS
SE_AS_LI_PA,   //SET ASYNC LINE PARAMETERS
SE_BO_MO,      //SET BOOTP MODE
SE_BO_SE_MO,   //SET BOOTP SERVER MODE
SE_BR_CO_MO,   //SET BRIDGE COMPRESSION MODE
SE_BR_FI_DE,   //SET BRIDGE FILTER DEFAULT
SE_BR_FI_MO,   //SET BRIDGE FILTER MODE
SE_BR_LE_MO,   //SET BRIDGE LEARNING MODE
SE_BR_MO,      //SET BRIDGE MODE
SE_BR_PO_CO,   //SET BRIDGE PORT COUNT
SE_BR_PO_ST_2, //SET BRIDGE PORT STATUS
SE_BR_PO_TR,   //SET BRIDGE PORT TRANSLATION
SE_COM_ME_LE,  //SET COMPRESSION MESSAGE LEVEL
SE_CONF,       //SET CONFIGURATION
SE_CONS_SP,    //SET CONSOLE SPEED
SE_DA,  //SET DATE
SE_DI_BA_CA_AD,        //SET DIAL BACKUP CALL ADDRESS
```

```
SE_DI_BA_CA_ST,        //SET DIAL BACKUP CALL STATE
SE_DI_BA_CO_TH,        //SET DIAL BACKUP CONGESTION THRESHOLDS
SE_DI_BA_DA,   //SET DIAL BACKUP DAMPING
SE_DI_BA_ER_TH,        //SET DIAL BACKUP ERROR THRESHOLD
SE_DI_BA_PO,   //SET DIAL BACKUP PORT
SE_DI_BA_RE_IN,        //SET DIAL BACKUP RETRY INTERVAL
SE_DI_BA_ST_TY,        //SET DIAL BACKUP STATION TYPE
SE_DI_PO_AD_ST,        //SET DIAL PORT ADMIN STATE
SE_DI_PO_AUTH_ME,      //SET DIAL PORT AUTHENTICATION METHOD
SE_DI_PO_CA_BA_ST,     //SET DIAL PORT CALLBACK STATUS
SE_DI_PO_CA_ST,        //SET DIAL PORT CALL STATE
SE_DI_PO_CL_IN,        //SET DIAL PORT CLEAR INTERVAL
SE_DI_PO_CO,   //SET DIAL PORT COUNT
SE_DI_PO_COM_ME,       //SET DIAL PORT COMPRESSION METHOD
SE_DI_PO_COM_ME_LE,    //SET DIAL PORT COMPRESSION MESSAGE LEVEL
SE_DI_PO_ME_LE,        //SET DIAL PORT MESSAGE LEVEL
SE_DI_PO_PR,   //SET DIAL PORT PRIORITY
SE_DI_PO_PRO,  //SET DIAL PORT PROTOCOL
SE_DI_PO_RE,   //SET DIAL PORT RETRY
SE_DI_PO_SE_TI,        //SET DIAL PORT SESSION TIMEOUT
SE_DI_PO_ST_TY,        //SET DIAL PORT STATION TYPE
SE_ET_MA_AD,   //SET ETHERNET MAC ADDRESS
SE_FDB_MA_SI,  //SET FDB MAXIMUM SIZE
SE_FDB_TI,     //SET FDB TIMEOUT
SE_FF_AD_FO_2, //SET FFR ADDRESS FORMAT
SE_FF_AD_LE_3, //SET FFR ADDRESS LENGTH
SE_FF_CIRCUIT_CI,      //SET FFR CIRCUIT CIR
SE_FF_CIRCUIT_CO_BU,   //SET FFR CIRCUIT COMMITTED BURST
SE_FF_CIRCUIT_EX_BU,   //SET FFR CIRCUIT EXCESS BURST
SE_FF_CIRCUIT_LO,      //SET FFR CIRCUIT LOOPBACK
SE_FF_CIRCUIT_ST,      //SET FFR CIRCUIT STATE
SE_FF_DL_4,    //SET FFR DLCMI
SE_FF_EN,      //SET FFR ENCAPSULATION
SE_FF_FU_7,    //SET FFR FULL_STAT_ENQ
SE_FF_ID_9,    //SET FFR IDLE_TIMER
SE_FF_MO_11,   //SET FFR MONITORING
SE_FF_PO_13,   //SET FFR POLL_INTERVAL
SE_FF_ST,      //SET FFR STATION_TYPE
SE_FI_BA,      //SET FILE BACKUP
SE_FI_CO,      //SET FILE COPY
SE_FI_DE,      //SET FILE DELETE
SE_FI_PR,      //SET FILE PRIMARY
SE_FI_RE,      //SET FILE RENAME
SE_FR_SW_ST,   //SET FRAME_RELAY SWITCH STATUS
SE_IP_AS_NU,   //SET IP AUTONOMOUS SYSTEM NUMBER
SE_IP_CI_RO_AP,        //SET IP ROUTE APPLICATION_ID
SE_IP_FO_ME_LE,        //SET IP MESSAGE LEVEL
SE_IP_NE_BR,   //SET IP NETWORK BROADCAST
SE_IP_NE_EN_TY,        //SET IP NETWORK ENTRY TYPE
SE_IP_NE_ME_1, //SET IP NETWORK METRIC
SE_IP_NE_MT_3, //SET IP NETWORK MTU
SE_IP_NE_SE_5, //SET IP NETWORK SECURITY
SE_IP_RO_CA_RE,        //SET IP ROUTE CACHE RESET
SE_IP_RO_ID,   //SET IP ROUTERID
SE_IP_RO_NO_ID,        //SET IP ROUTE NOTIFICATION IDS
SE_IP_SO_RO,   //SET IP SOURCE ROUTING
SE_IP_UN_DE_SO_AD,     //SET IP UNNUMBERED INTERFACE DEFAULT SOURCE ADDRESS
SE_IP_UN_IN_SO_AD,     //SET IP UNNUMBERED INTERFACE SOURCE ADDRESS
SE_LA_CL_MO,   //SET LAPB CLOCK MODE
SE_LA_FC,      //SET LAPB FCS
SE_LA_FL,      //SET LAPB FLAGS
SE_LA_FR_WI,   //SET LAPB FRAME WINDOW
SE_LA_N2,      //SET LAPB N2
SE_LA_ST_TY,   //SET LAPB STATION TYPE
SE_LA_T1,      //SET LAPB T1
SE_LA_T3,      //SET LAPB T3
SE_MC_EXEC,    //SET MACRO EXECUTE
SE_ME_DB_IN,   //SET MEMORY DBLOCK INCREMENT
SE_ME_OP_FE,   //SET MEMORY OPTIMIZER FEATURE
SE_ME_OP_ST,   //SET MEMORY OPTIMIZER STATE
SE_MU_GR_AD_ST,        //SET MULTILINK GROUP ADMIN STATUS
SE_MU_GR_BA_MA,        //SET MULTILINK GROUP BANDWIDTH MANAGEMENT
SE_MU_GR_CO,   //SET MULTILINK GROUP COUNT
```

```
SE_MU_GR_CO_ME,        //SET MULTILINK GROUP COMPRESSION METHOD
SE_MU_GR_CO_ME_LE,     //SET MULTILINK GROUP COMPRESSION MESSAGE LEVEL
SE_MU_GR_CO_TH,        //SET MULTILINK GROUP CONGESTION THRESHOLDS
SE_MU_GR_DA,   //SET MULTILINK GROUP DAMPING
SE_MU_GR_FR,   //SET MULTILINK GROUP FRAGMENTATION
SE_MU_GR_ME_LE,        //SET MULTILINK GROUP MESSAGE LEVEL
SE_MU_GR_PR,   //SET MULTILINK GROUP PROTOCOL
SE_PH_PO_CO_NOREV,     //SET PHYSICAL PORT COMPRESSION
SE_PH_PO_DI_AD,        //SET PHYSICAL PORT DIAL ADDRESS
SE_PH_PO_DI_PR,        //SET PHYSICAL PORT DIAL PROCEDURE
SE_PH_PO_E1_FR_MO,     //SET PHYSICAL PORT E1 FRAME MODE
SE_PH_PO_NA,   //SET PHYSICAL PORT NAME
SE_PH_PO_PR_SD,        //SET PHYSICAL PORT PROTOCOL
SE_PH_PO_QU,   //SET PHYSICAL PORT QUEUE
SE_PH_PO_RE_MO,        //SET PHYSICAL PORT RESYNC MODE
SE_PH_PO_SC,   //SET PHYSICAL PORT SCT
SE_PH_PO_SP,   //SET PHYSICAL PORT SPEED
SE_PH_PO_ST,   //SET PHYSICAL PORT STATUS
SE_PI_PA,      //SET PING PARAMETERS
SE_PP_AC_OP,   //SET PPP ACCOUNTING OPTION
SE_PP_AU_IN,   //SET PPP AUTHENTICATION IN
SE_PP_AU_OU,   //SET PPP AUTHENTICATION OUT
SE_PP_AU_RE,   //SET PPP AUTHENTICATION RETRY
SE_PP_BR_ST,   //SET PPP BRIDGE STATE
SE_PP_COM_ME_LE,       //SET PPP COMPRESSION MESSAGE LEVEL
SE_PP_CO_ME,   //SET PPP COMPRESSION METHOD
SE_PP_IP_ST,   //SET PPP IP STATE
SE_PP_LI_ST,   //SET PPP LINK STATE
SE_PP_MA_CO,   //SET PPP MAXIMUM CONFIGURE
SE_PP_MA_FA,   //SET PPP MAXIMUM FAILURE
SE_PP_MA_TE,   //SET PPP MAXIMUM TERMINATE
SE_PP_ME_LE_90,        //SET PPP MESSAGE LEVEL
SE_PP_MO,      //SET PPP MONITORING
SE_PP_PO_IN,   //SET PPP POLL INTERVAL
SE_PP_RE_TI,   //SET PPP RESTART TIMER
SE_PROM,       //SET PROMPT
SE_PROT_PR_DE, //SET PROTOCOL PRIORITY DEFAULT
SE_PR_IP_FI_AP,        //SET PROFILE IP FILTER APPLICATION_ID
 SE_SC_CO,       //SET SCRIPT CONTINUATION
 SE_SC_HA,       //SET SCRIPT HALT
 SE_SC_VE,       //SET SCRIPT VERSION
SE_SE_WI_LE,   //SET SESSION WINDOW LENGTH
SE_SE_WI_WI,   //SET SESSION WINDOW WIDTH
SE_SN_AU_TR_MO,        //SET SNMP AUTHENTICATION TRAP MODE
SE_STA_DI,     //SET STATISTICS DISPLAY
SE_STP_BR_PR,  //SET STP BRIDGE PRIORITY
SE_STP_FO_DE,  //SET STP FORWARD DELAY
SE_STP_HE_TI,  //SET STP HELLO TIME
SE_STP_MA_AG,  //SET STP MAXIMUM AGE
SE_STP_MO,     //SET STP MODE
SE_STP_MU,     //SET STP MULTICAST
SE_STP_PO_PA_CO,       //SET STP PORT PATH COST
SE_STP_PO_PR,  //SET STP PORT PRIORITY
SE_STP_PO_ST,  //SET STP PORT STATE
SE_SY_CO,      //SET SYSTEM CONTACT
SE_SY_DE,      //SET SYSTEM DEBUG
SE_SY_DO,      //SET SYSTEM DOMAIN
SE_SY_LO,      //SET SYSTEM LOCATION
SE_SY_MA_AD,   //SET SYSTEM MAC ADDRESS
SE_SY_ME_LE,   //SET SYSTEM MESSAGE LEVEL
SE_SY_OP_FE,   //SET SYSTEM OPTIMIZER FEATURE
SE_SY_OP_ST,   //SET SYSTEM OPTIMIZER STATE
SE_SY_SE_PO,   //SET SYSTEM SERVICE PORT
SE_TI,  //SET TIMEZONE
SE_TR_LO_BU_SI,        //SET TRAP LOG BUFFER SIZE
SE_TR_LO_LE,   //SET TRAP LOG LEVEL
SE_TR_LO_SA,   //SET TRAP LOG SAVE
SE_UDP_HE_ST,  //SET UDP HELPER STATUS
SHOW,   //SHOW
SNMPGET,       //SNMPGET
SNMPGETN,      //SNMPGETNEXT
SNMPS_INT,     //SNMPSET INTEGER
SNMPS_NET,     //SNMPSET NETADDR
```

```
                SNMPS_OCT,     //SNMPSET OCTETS
                TELNET, //TELNET
                TF_FS,  //TFTP
                TR_RT,  //TRACEROUTE
            }
      Agents { REUTERS_CMDS }   // TELNET, only support ourself
   <\Product>

   <Objects>

          ObjectName            { oReutersClearAlarm } {
                ObjectId        { 43.6.1.4.1.5.2.1.4.1 }
                Title           { Clear Alarm }
                Format          {       <Enumerations>
                                        ON(1),
                                        OFF(2)
                                        <\Enumerations>
                                }
                // Fieldwidth defaults to the title_len + 1or2
          }

          ObjectName            { oReutersClearStats } {
                ObjectId        { 43.6.1.4.1.5.2.1.4.2 }
                // Format defaults to F_GENERIC
                Title           { Clear Stats }
                // Fieldwidth defaults to the title_len + 1o r2
          }

          ObjectName            { oReutersBrStat } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.13 }
                // Format defaults to F_GENERIC
                Title           { BrStat }
                // Fieldwidth defaults to the title_len + 1or2
          }

          ObjectName            { oReutersBrType } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.12 }
                Title           { Bridge Type }
                Format          {       <Enumerations>
                                        NONE(0),
                                        NORTH(1),
                                        SOUTH(2)
                                        <\Enumerations>
                                }
          }

          ObjectName            { oReutersMonHelloPoll } {
                ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.13 }
                // Format defaults to F_GENERIC
                Title           { Hello Poll }
                // Fieldwidth defaults to the title_len + 1or2
          }

          ObjectName            { oReutersMonBufferHighWater } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.5 }
                // Format defaults to F_GENERIC
                Title           { MEM U-High Thresh }
                // Fieldwidth defaults to the title_len + 1or2
          }

          ObjectName            { oReutersMonBufferLowWater } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.6 }
                // Format defaults to F_GENERIC
                Title           { MEM U-Low Thresh }
                // Fieldwidth defaults to the title_len + 1or2
          }

          ObjectName            { oReutersCpuThrLoad } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.11 }
                // Format defaults to F_GENERIC
                Title           { CPU U-Thresh }
                // Fieldwidth defaults to the title_len + 1or2
          }
```

```
ObjectName            { oReutersMonEnetDeletedThreshold } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.5 }
       // Format defaults to F_GENERIC
       Title          { LAN Deleted }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonEnetInterPacketTime } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.7 }
       // Format defaults to F_GENERIC
       Title          { LAN I-Packet Time }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonEnetTxThreshold } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.6 }
       // Format defaults to F_GENERIC
       Title          { LAN Tx Error }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonHDLCQualityThreshold } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.2 }
       // Format defaults to F_GENERIC
       Title          { WAN Err Threshold }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonHDLCQualitySample } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.12 }
       // Format defaults to F_GENERIC
       Title          { WAN Err Sample }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonHDLCInactivity } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.3 }
       Title          { WAN Inactivity }
       // Fieldwidth defaults to the title_len + 1or2
       Format                {      <Enumerations>
                                    ON(1),
                                    OFF(2),
                                    N\/A(3)
                                    <\Enumerations>
                             }
}

ObjectName            { oReutersMonLAPBTransit_Multi } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.7 }
       // Format defaults to F_GENERIC
       Title          { WAN T-Multi }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonLAPBUHighWater } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.5 }
       // Format defaults to F_GENERIC
       Title          { WAN U-Thresh }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonLAPBUSpeed } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.4 }
       // Format defaults to F_GENERIC
       Title          { WAN U-Speed }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonLAPB2lq } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.8 }
       // Format defaults to F_GENERIC
       Title          { WAN 2LQ }
```

```
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonLAPBTestTMO } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.8.1.10 }
        // Format defaults to F_GENERIC
        Title           { WAN Test TMO }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonLAPBT1Variance } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.8.1.11 }
        // Format defaults to F_GENERIC
        Title           { T1 Variance }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonMLinkLost } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.4 }
        // Format defaults to F_GENERIC
        Title           { WAN Multilink Lost }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonMLinkPackingTMO } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.9 }
        // Format defaults to F_GENERIC
        Title           { Packing TMO }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonMLinkPackingType } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.8 }
        // Format defaults to F_GENERIC
        Title           { Packing Type }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonMLinkDelayDelta } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.10 }
        // Format defaults to F_GENERIC
        Title           { Delay Delta }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonAlarmIndex } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.3.1.1 }
        Format          { F_INT_IF2NAME }
        Title           { Port }
        // Fieldwidth defaults to the title_len + 1or2
}
ObjectName              { oReutersMonHDLCInactivityAlarm } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.3.1.3 }
        Format          {       <Enumerations>
                                ON(1),
                                OFF(2),
                                N\/A(3)
                                <\Enumerations>
                        }
        Title           { WAN Inactivity }
        // Fieldwidth defaults to the title_len + 1or2
}
ObjectName              { oReutersMonHDLCQualityAlarm } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.3.1.2 }
        Format          {       <Enumerations>
                                ON(1),
                                OFF(2),
                                N\/A(3)
                                <\Enumerations>
                        }
        Title           { WAN Error }
        // Fieldwidth defaults to the title_len + 1or2
}
```

```
ObjectName            { oReutersMonLAPBUCurrent } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.2 }
       // Format defaults to F_GENERIC
       Title          { WAN U-Current }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonLAPBUHigh } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.3 }
       // Format defaults to F_GENERIC
       Title          { WAN U-Highest }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonLAPBTransit_Delay } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.6 }
       // Format defaults to F_GENERIC
       Title          { WAN T-Delay }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonEnetDeletedAlarm } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.3.1.5 }
       Title          { LAN Deleted }
       Format         {      <Enumerations>
                             ON(1),
                             OFF(2),
                             N\/A(3)
                             <\Enumerations>
                      }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonEnetTxAlarm } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.3.1.6 }
       Title          { LAN Tx Error }
       Format         {      <Enumerations>
                             ON(1),
                             OFF(2),
                             N\/A(3)
                             <\Enumerations>
                      }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonEnetInterPacketTimeAlarm } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.3.1.7 }
       Title          { LAN I-Packet Time }
       Format         {      <Enumerations>
                             ON(1),
                             OFF(2),
                             N\/A(3)
                             <\Enumerations>
                      }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonMLinkLostAlarm } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.3.1.4 }
       Title          { WAN Multilink Lost }
       Format         {      <Enumerations>
                             ON(1),
                             OFF(2),
                             N\/A(3)
                             <\Enumerations>
                      }
       // Fieldwidth defaults to the title_len + 1or2
}

ObjectName            { oReutersMonAvgPktSz } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.8.1.9 }
       // Format defaults to F_GENERIC
       Title          { Avg Pkt Siz }
       // Fieldwidth defaults to the title_len + 1or2
}
ObjectName            { oReutersMonMLinkState } {
       ObjectId       { 43.6.1.4.1.5.2.1.1.2.1.11 }
       Title          { WAN State }
       Format         {      <Enumerations>
```

```
                                DOWN(0),
                                SETUP(1),
                                UP(2),
                                STDBY(3)
                                <\Enumerations>
                        }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonParmIndex } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.2.1.1 }
        Format          { F_INT_IF2NAME }
        Title           { Port }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonAlarm } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.4 }
        Title           { Alarm Status }
        Format          {       <Enumerations>
                                ON(1),
                                OFF(2)
                                <\Enumerations>
                        }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonAll } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.5 }
        Title           { System Status }
        // Format defaults to F_GENERIC
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonAllString } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.6 }
        Title           { System Status }
        Format          {       <Enumerations>
                                LIVE\/NO\ ALARM(1),
                                LIVE\/ALARM(2),
                                STANDBY\/ALARM(3),
                                STANDBY\/NO\ ALARM(4)
                                <\Enumerations>
                        }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonLiveStandby } {
        ObjectId        { 43.6.1.4.1.5.2.1.1.7 }
        Title           { Operational Status }
        Format          {       <Enumerations>
                                LIVE(0),
                                STANDBY(1)
                                <\Enumerations>
                        }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonBufferCurrent } {
        ObjectId        { 43.6.1.4.1.5.2.1.3.8 }
        // Format defaults to F_GENERIC
        Title           { MEM U-Current }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersMonBufferHighest } {
        ObjectId        { 43.6.1.4.1.5.2.1.3.7 }
        // Format defaults to F_GENERIC
        Title           { MEM U-Highest }
        // Fieldwidth defaults to the title_len + 1or2
}

ObjectName              { oReutersLowPriorityDiscards } {
```

```
                ObjectId        { 43.6.1.4.1.5.2.1.3.14 }
                // Format defaults to F_GENERIC
                Title           { Low Prior Disc }
                // Fieldwidth defaults to the title_len + 1or2
        }

        ObjectName              { oReutersCpuAvgLoad } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.9 }
                // Format defaults to F_GENERIC
                Title           { CPU U-Current }
                // Fieldwidth defaults to the title_len + 1or2
        }

        ObjectName              { oReutersCpuMaxLoad } {
                ObjectId        { 43.6.1.4.1.5.2.1.3.10 }
                // Format defaults to F_GENERIC
                Title           { CPU U-Highest }
                // Fieldwidth defaults to the title_len + 1or2
        }

        ObjectName            { oFdbPriorityRangeEntSrcBegMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.3.1.1.1.0 }
               Format         { PSTYPMACADDR }
               Title          { Begin MAC Address }
        }
        ObjectName            { oFdbPriorityRangeEntSrcEndMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.3.1.1.2 }
               Format         { PSTYPMACADDR }
               Title          { End MAC Address }
        }

        ObjectName            { oFdbPriorityRangeEntDstBegMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.3.2.1.1 }
               Format         { PSTYPMACADDR }
               Title          { Begin MAC Address }
        }
        ObjectName            { oFdbPriorityRangeEntDstEndMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.3.2.1.2 }
               Format         { PSTYPMACADDR }
               Title          { End MAC Address }
        }

        ObjectName            { oFdbRangeEntBegMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.2.1.1.1 }
               Format         { PSTYPMACADDR }
               Title          { Begin MAC Address }
        }
        ObjectName            { oFdbRangeEntEndMacAddr } {
               ObjectId       { 43.6.1.4.1.5.2.1.2.1.1.2 }
               Format         { PSTYPMACADDR }
               Title          { End MAC Address }
        }
        ObjectName            { oFdbRangeEntDisp } {
               ObjectId       { 43.6.1.4.1.5.2.1.2.1.1.3 }
               Format         {      <Enumerations>
                                     FORWARD(0),
                                     FLOOD(1),
                                     DISCARD(2),
                                     <\Enumerations>
                              }
               Title          { Disposition }
               // Fieldwidth defaults to the title_len + 1or2
        }
        ObjectName            { oFdbRangeEntPort } {
               ObjectId       { 43.6.1.4.1.5.2.1.2.1.1.4 }
               Format         { F_GENERIC }
               Title          { Port }
               // Fieldwidth defaults to the title_len + 1or2
        }

<\Objects>
<Commands>
```

```
CommandId             { CL_MO_AL } {
       Keywords       { CLEAR MONITOR ALARM }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersClearAlarm }
              parmtype              { PSTYPINT }
              parmflags             {
                                    forcedef(2)
                                    }
              in_instance           { no }
              },
       <\Parameters>
}

CommandId             { CL_MO_ST } {
       Keywords       { CLEAR MONITOR STATS }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersClearStats }
              parmtype              { PSTYPINT }
              parmflags             {
                                    forcedef(2)
                                    }
              in_instance           { no }
              },
       <\Parameters>
}

CommandId             { SE_BR_ST } {
       Keywords       { SET BRIDGE STANDBY }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersBrStat }
              parmtype              { PSTYPINT }
              parmflags             {
                                    forcedef(2)
                                    }
              in_instance           { no }
              },
       <\Parameters>
}

CommandId             { SE_BR_TY } {
       Keywords       { SET BRIDGE TYPE }
       Function       { nc_SetBridge }
       <Parameters>
              {
              object_name           { oReutersBrType }
              parmtype              { PSTYPSTRCV }
              helpstring            { TYPE }
              parmflags             {
                                    minimum(1),
                                    maximum(2)
                                    }
              in_instance           { no }
              },
              {
              object_name           { NULL }
              parmtype              { PSTYPINT }
              helpstring            { COUNT [1 - 4] }
              parmflags             {
                                    minimum(1),
                                    maximum(4),
                                    optional(4)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}
```

```
CommandId             { SE_MO_BU_HI_TH } {
       Keywords       { SET MONITOR BUFFER HIGH THRESHOLD }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersMonBufferHighWater }
              parmtype              { PSTYPINT }
              helpstring            { PERCENT [0 - 100] }
              parmflags             {
                                    minimum(0),
                                    maximum(100)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_BU_LO_TH } {
       Keywords       { SET MONITOR BUFFER LOW THRESHOLD }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersMonBufferLowWater }
              parmtype              { PSTYPINT }
              helpstring            { PERCENT [0 - 100] }
              parmflags             {
                                    minimum(0),
                                    maximum(100)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_CP_TH } {
       Keywords       { SET MONITOR CPU_LOAD THRESHOLD }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oReutersCpuThrLoad }
              parmtype              { PSTYPINT }
              helpstring            { PERCENT [0 - 100] }
              parmflags             {
                                    minimum(0),
                                    maximum(100)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_HE } {
       Keywords       { SET MONITOR HELLO_POLL }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oIfIndex }
              parmtype              { PSTYPPORT }
              helpstring            { PORT_ID [PHYSICAL] }
              in_instance           { yes }
              },
              {
              object_name           { oReutersMonHelloPoll }
              parmtype              { PSTYPINT }
              helpstring            { COUNT [0 - 65535] }
              parmflags             {
                                    minimum(0),
                                    maximum(65535)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}
```

```
CommandId             { SE_MO_EN_DE_TH } {
       Keywords       { SET MONITOR ENET DELETED THRESHOLD }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oIfIndex }
              parmtype              { PSTYPPORT }
              helpstring            { PORT_ID [PHYSICAL] }
              in_instance           { yes }
              },
              {
              object_name           { oReutersMonEnetDeletedThreshold }
              parmtype              { PSTYPINT }
              helpstring            { COUNT [0 - 65535] }
              parmflags             {
                                    minimum(0),
                                    maximum(65535)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_EN_IN_TI } {
       Keywords       { SET MONITOR ENET INTERPACKET TIME }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oIfIndex }
              parmtype              { PSTYPPORT }
              helpstring            { PORT_ID [PHYSICAL] }
              in_instance           { yes }
              },
              {
              object_name           { oReutersMonEnetInterPacketTime }
              parmtype              { PSTYPINT }
              helpstring            { SECONDS [0, 10 - 6000] }
              parmflags             {
                                    minimum(0),
                                    maximum(6000)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_EN_TR_TH } {
       Keywords       { SET MONITOR ENET TRANSMIT_ERRORS THRESHOLD }
       Function       { nc_genSet }
       <Parameters>
              {
              object_name           { oIfIndex }
              parmtype              { PSTYPPORT }
              helpstring            { PORT_ID [PHYSICAL] }
              in_instance           { yes }
              },
              {
              object_name           { oReutersMonEnetTxThreshold }
              parmtype              { PSTYPINT }
              helpstring            { COUNT [0 - 65535] }
              parmflags             {
                                    minimum(0),
                                    maximum(65535)
                                    }
              in_instance           { no }
              },
              <\Parameters>
}

CommandId             { SE_MO_LA_ER_TH } {
       Keywords       { SET MONITOR LAPB ERROR THRESHOLD }
       Function       { nc_genSet }
```

```
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonHDLCQualityThreshold }
                parmtype                { PSTYPINT }
                helpstring              { PERCENT [0 - 100] }
                parmflags               {
                                        minimum(0),
                                        maximum(100)
                                        }
                in_instance             { no }
                },
                {
                object_name             { oReutersMonHDLCQualitySample }
                parmtype                { PSTYPINT }
                helpstring              { COUNT [0 - 65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_IN } {
        Keywords        { SET MONITOR LAPB INACTIVITY }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonHDLCInactivity }
                parmtype                {       <InputEnumerations>
                                                ON(1),
                                                OFF(2)
                                                <\InputEnumerations>
                                        }
                helpstring              { MONITOR_MODE }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_TR } {
        Keywords        { SET MONITOR LAPB TRANSIT_MULTI }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPBTransit_Multi }
                parmtype                { PSTYPINT }
                helpstring              { COUNT [2 - 16000] }
                parmflags               {
                                        minimum(2),
                                        maximum(16000)
                                        }
                in_instance             { no }
                },
```

```
                <\Parameters>
}

CommandId               { SE_MO_LA_UT_PE } {
        Keywords        { SET MONITOR LAPB UTILIZATION PERCENT }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPBUHighWater }
                parmtype                { PSTYPINT }
                helpstring              { PERCENT [0 - 100] }
                parmflags               {
                                        minimum(0),
                                        maximum(100)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_UT_SP } {
        Keywords        { SET MONITOR LAPB UTILIZATION SPEED }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPBUSpeed }
                parmtype                { PSTYPINT }
                helpstring              { RATE [0 - 2048000000] }
                parmflags               {
                                        minimum(0),
                                        maximum(2048000000)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_2L } {
        Keywords        { SET MONITOR LAPB 2LQ }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPB2lq }
                parmtype                { PSTYPINT }
                helpstring              { BYTES [0 - 65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_TE } {
```

```
        Keywords        { SET MONITOR LAPB TESTTMO }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPBTestTMO }
                parmtype                { PSTYPINT }
                helpstring              { 1/10 SEC [10 - 600] }
                parmflags               {
                                        minimum(10),
                                        maximum(600)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_LA_T1_VA } {
        Keywords        { SET MONITOR LAPB T1 VARIANCE }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonLAPBT1Variance }
                parmtype                { PSTYPINT }
                helpstring              { MSEC [0-65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_ML_LO_TH } {
        Keywords        { SET MONITOR MLINK LOST THRESHOLD }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonMLinkLost }
                parmtype                { PSTYPINT }
                helpstring              { COUNT [0 - 65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_ML_PA_TM } {
        Keywords        { SET MONITOR MLINK PACKING TMO }
        Function        { nc_genSet }
        <Parameters>
                {
```

```
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonMLinkPackingTMO }
                parmtype                { PSTYPINT }
                helpstring              { PACKING_TMO [0-65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_ML_PA_TY } {
        Keywords        { SET MONITOR MLINK PACKING TYPE }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonMLinkPackingType }
                parmtype                { PSTYPINT }
                helpstring              { PACKING_TYPE [0-2] }
                parmflags               {
                                        minimum(0),
                                        maximum(2)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_ML_DE_DE } {
        Keywords        { SET MONITOR MLINK DELAY DELTA }
        Function        { nc_genSet }
        <Parameters>
                {
                object_name             { oIfIndex }
                parmtype                { PSTYPPORT }
                helpstring              { PORT_ID [PHYSICAL] }
                in_instance             { yes }
                },
                {
                object_name             { oReutersMonMLinkDelayDelta }
                parmtype                { PSTYPINT }
                helpstring              { MSEC [0-65535] }
                parmflags               {
                                        minimum(0),
                                        maximum(65535)
                                        }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { SE_MO_ST } {
        Keywords        { SET MONITOR STATUS }
        Function         { nc_SetMonStat }
        <Parameters>
                {
                object_name             { oDummy }
                parmtype                {       <InputEnumerations>
                                                ON(1),
                                                OFF(2),
```

```
                                                ANSI(3)
                                                <\InputEnumerations>
                                        }
                helpstring              { STATUS }
                in_instance             { no }
                },
                <\Parameters>
}

CommandId               { DI_MON_ST_TA } {
        Keywords        { DISPLAY MONITOR STATUS TABLE }
        Function        { nc_genGNext }
        DisplayObjects  {
                        oReutersMonAlarmIndex,
                        oReutersMonHDLCInactivityAlarm,
                        oReutersMonHDLCQualityAlarm,
                        oReutersMonLAPBUCurrent,
                        oReutersMonLAPBUHigh,
                        oReutersMonLAPBTransit_Delay,
                        oReutersMonEnetDeletedAlarm,
                        oReutersMonEnetTxAlarm,
                        oReutersMonEnetInterPacketTimeAlarm,
                        oReutersMonMLinkLostAlarm,
                        oReutersMonAvgPktSz,
                        oReutersMonMLinkState
                        }
        Display         { CFLG_TABLE3COL }
}

CommandId               { DI_MO_PA_TA } {
        Keywords        { DISPLAY MONITOR PARAMETER TABLE }
        Function        { nc_genGNext }
        DisplayObjects  {
                        oReutersMonParmIndex,
                        oReutersMonHDLCInactivity,
                        oReutersMonHDLCQualityThreshold,
                        oReutersMonHDLCQualitySample,
                        oReutersMonLAPBUHighWater,
                        oReutersMonLAPBUSpeed,
                        oReutersMonLAPBTransit_Multi,
                        oReutersMonLAPB2lq,
                        oReutersMonLAPBTestTMO,
                        oReutersMonEnetDeletedThreshold,
                        oReutersMonEnetTxThreshold,
                        oReutersMonEnetInterPacketTime,
                        oReutersMonMLinkLost,
                        oReutersMonMLinkDelayDelta,
                        oReutersMonMLinkPackingType,
                        oReutersMonMLinkPackingTMO,
                        oReutersMonHelloPoll,
                        oReutersMonLAPBT1Variance
                        }
        Display         { CFLG_TABLE3COL }
}

CommandId               { DI_MO_SY_IN } {
        Keywords        { DISPLAY MONITOR SYSTEM INFO }
        Function        { nc_StatMode }
        DisplayObjects  {
                        oReutersMonAlarm,
                        oReutersMonLiveStandby,
                        oReutersMonBufferLowWater,
                        oReutersMonBufferHighWater,
                        oReutersMonBufferCurrent,
                        oReutersMonBufferHighest,
                        oReutersLowPriorityDiscards,
                        oReutersCpuThrLoad,
                        oReutersCpuAvgLoad,
                        oReutersCpuMaxLoad
                        }
        StatTab         { OR_MONSTATUSTAB }
        Display         { CFLG_NONE }
}
```

```
CommandId             { SE_MO_UP_IN } {
       Keywords       { SET MONITOR UPDATE INTERVAL }
       Function       { nc_SetMonUpdateInt }
       <Parameters>
              {
              object_name    { NULL }
              parmtype       { PSTYPINT }
              parmflags      {
                             minimum(5),
                             maximum(65535),
                             optional(10)
                             }
              helpstring     { COUNT [5 - 65535] }
              in_instance    { no }
              },
       <\Parameters>
}

CommandId             { DI_MO_UP_IN } {
       Keywords       { DISPLAY MONITOR UPDATE INTERVAL }
       Function       { nc_DisMonUpdateInt }
       Display        { CFLG_NONE }
}

CommandId {AD_FD_SO_PR_RA_EN } {
Keywords { ADD FDB SOURCE PRIORITY RANGE ENTRY }
Function { nc_genSet }
<Parameters>
       {      object_name    { oFdbPriorityRangeEntSrcBegMacAddr }
              parmtype       { PSTYPMACADDR }
              helpstring     { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
              in_instance    { yes }
       },
       {      object_name    { oFdbPriorityRangeEntSrcEndMacAddr }
              parmtype       { PSTYPMACADDR }
              helpstring     { END_MAC_ADDRESS [A:B:C:D:E:F] }
              in_instance    { no }
       }
<\Parameters>
}

CommandId { DE_FD_SO_PR_RA_EN } {
Keywords { DELETE FDB SOURCE PRIORITY RANGE ENTRY }
Function { nc_genSet }
<Parameters>
       {      object_name    { oFdbPriorityRangeEntSrcBegMacAddr }
              parmtype       { PSTYPMACADDR }
              helpstring     { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
              in_instance    { yes }
       }
<\Parameters>
}

CommandId             { DI_FD_SO_PR_RA_TA } {
       Keywords       { DISPLAY FDB SOURCE PRIORITY RANGE TABLE }
       Function       { nc_genGNext }
       DisplayObjects {
                      oFdbPriorityRangeEntSrcBegMacAddr,
                      oFdbPriorityRangeEntSrcEndMacAddr
                      }
       StatTab        { OR_FSPRT }
       Display        { CFLG_TABLE }
}

CommandId {AD_FD_DE_PR_RA_EN } {
Keywords { ADD FDB DESTINATION PRIORITY RANGE ENTRY }
Function { nc_genSet }
<Parameters>
       {      object_name    { oFdbPriorityRangeEntDstBegMacAddr }
              parmtype       { PSTYPMACADDR }
              helpstring     { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
              in_instance    { yes }
```

```
        },
        {       object_name     { oFdbPriorityRangeEntDstEndMacAddr }
                parmtype        { PSTYPMACADDR }
                helpstring      { END_MAC_ADDRESS [A:B:C:D:E:F] }
                in_instance     { no }
        }
<\Parameters>
}

CommandId { DE_FD_DE_PR_RA_EN } {
Keywords { DELETE FDB DESTINATION PRIORITY RANGE ENTRY }
Function { nc_genSet }
<Parameters>
        {       object_name     { oFdbPriorityRangeEntDstBegMacAddr }
                parmtype        { PSTYPMACADDR }
                helpstring      { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
                in_instance     { yes }
        }
<\Parameters>
}

CommandId               { DI_FD_DE_PR_RA_TA } {
        Keywords        { DISPLAY FDB DESTINATION PRIORITY RANGE TABLE }
        Function        { nc_genGNext }
        DisplayObjects {
                        oFdbPriorityRangeEntDstBegMacAddr,
                        oFdbPriorityRangeEntDstEndMacAddr
                        }
        StatTab         { OR_FDPRT }
        Display         { CFLG_TABLE }
}

CommandId {AD_FD_RA_EN } {
Keywords { ADD FDB RANGE ENTRY }
Function { nc_genSet }
<Parameters>
        {       object_name     { oFdbRangeEntBegMacAddr }
                parmtype        { PSTYPMACADDR }
                helpstring      { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
                in_instance     { yes }
        },
        {       object_name     { oFdbRangeEntEndMacAddr }
                parmtype        { PSTYPMACADDR }
                helpstring      { END_MAC_ADDRESS [A:B:C:D:E:F] }
                in_instance     { no }
        },
        {       object_name     { oFdbRangeEntDisp }
                parmtype        { PSTYPSTRCV }
                helpstring      { DISPOSITION }
                in_instance     { no }
        },
        {       object_name     { oFdbRangeEntPort }
                parmtype        { PSTYPINT }
                helpstring      { PORT_ID [1-66] }
                parmflags       {
                                minimum(1),
                                maximum(66),
                                optional(2)
                                }
                in_instance     { no }
        }
<\Parameters>
}

CommandId { DE_FD_RA_EN } {
Keywords { DELETE FDB RANGE ENTRY }
Function { nc_genSet }
<Parameters>
        {       object_name     { oFdbRangeEntBegMacAddr }
                parmtype        { PSTYPMACADDR }
                helpstring      { BEG_MAC_ADDRESS [A:B:C:D:E:F] }
                in_instance     { yes }
        }
```

```
        <\Parameters>
        }

        CommandId             { DI_FD_RA_TA } {
               Keywords       { DISPLAY FDB RANGE TABLE }
               Function       { nc_genGNext }
               DisplayObjects {
                              oFdbRangeEntBegMacAddr,
                              oFdbRangeEntEndMacAddr,
                              oFdbRangeEntDisp,
                              oFdbRangeEntPort
                              }
               Display        { CFLG_TABLE }
        }

        CommandId             { DI_CONF_FS_REUTERS } {
               Keywords       { DISPLAY CONFIGURATION }
               Function       { nc_genDisp }
               DisplayObjects {
                              oEthNumber,
                              oHdlcNumber,
                              oRomId,
                              oSysDescr,
                              oSoftwareVersion,
                              oSysProcType,
                              oSysLocalRam,
                              oSysGlobalRam,
                              oSysPowerSupply,
                              oFsLoadFtkFile,
                              oFsLoadAplFile,
                              oFsLoadDiaFile,
                              oFsLoadConFile,
                              oFsLoadScrFile,
                              oSysMsgLevel
                              }
        }

        CommandId             { DI_BR_TY } {
               Keywords       { DISPLAY BRIDGE TYPE }
               Function       { nc_genDisp }
               DisplayObjects {
                              oReutersBrType
                              }
        }

<\Commands>
```

B.1 Reuters State Change Table

The following section is provided to clarify the various States and Events (conditions which cause the IDN BRIDGE to transition from one state to another state). The State Table also provides clarity on the correct action of the Live Signal, Live/Standby Status indication, Healthy Signal and Alarm Status indication. The initial state for a newly powered up IDN BRIDGE (and after a "reset") is STBY_NO_ALARM.

Current State	Event	Next State	Live/Standby Status	Alarm Status	Live Signal	Healthy Signal	Notes
STBY_NO_ALARM	Clear Alarm	Same	Standby	No Alarm	On	On	
STBY_NO_ALARM	Alarm Trigger	RDY_ALARM	Standby	Alarm	On	Off	1
STBY_NO_ALARM	Multilink Up	LIVE_NO_ALARM	Live	No Alarm	On	On	
STBY_NO_ALARM	Multilink Down	STBY_ALARM	Standby	Alarm	On	Off	2
STBY_NO_ALARM	Switch Trigger	STBY_ALARM	Standby	Alarm	Off	Off	
LIVE_NO_ALARM	Clear Alarm	Same	Live	No Alarm	On	On	
LIVE_NO_ALARM	Alarm Trigger	LIVE_ALARM	Live	Alarm	On	Off	
LIVE_NO_ALARM	Multilink Up	Same	Live	No Alarm	On	On	3
LIVE_NO_ALARM	Multilink Down	RDY_ALARM	Standby	Alarm	On	Off	
LIVE_NO_ALARM	Switch Trigger	STBY_ALARM	Standby	Alarm	Off	Off	
LIVE_ALARM	Clear Alarm	LIVE_NO_ALARM	Live	No Alarm	On	On	
LIVE_ALARM	Alarm Trigger	Same	Live	Alarm	On	Off	
LIVE_ALARM	Multilink Up	Same	Live	Alarm	On	Off	3
LIVE_ALARM	Multilink Down	RDY_ALARM	Standby	Alarm	On	Off	
LIVE_ALARM	Switch Trigger	STBY_ALARM	Standby	Alarm	Off	Off	
STBY_ALARM	Clear Alarm	STBY_NO_ALARM	Standby	No Alarm	On	On	
STBY_ALARM	Alarm Trigger	Same	Standby	Alarm	Off	Off	1
STBY_ALARM	Multilink Up	STBY_ALARM	Standby	Alarm	Off	Off	
STBY_ALARM	Multilink Down	Same	Standby	Alarm	Off	Off	2
STBY_ALARM	Switch Trigger	Same	Standby	Alarm	Off	Off	
RDY_ALARM	Clear Alarm	STBY_NO_ALARM	Standby	No Alarm	On	On	
RDY_ALARM	Alarm Trigger	Same	Standby	Alarm	On	Off	1
RDY_ALARM	Multilink Up	LIVE_ALARM	Live	Alarm	On	Off	
RDY_ALARM	Multilink Down	Same	Standby	Alarm	On	Off	2
RDY_ALARM	Switch Trigger	STBY_ALARM	Standby	Alarm	Off	Off	

Notes:

1. Since all WANs in the Multilink Group are inactive in this state, the only Alarm Trigger event which can happen is the LAN Frames Missed and Deleted Alarm.

2. Since all WANs in the Multilink Group are inactive in this state, the Multilink Down event can never happen since it is already down.

3. Since at least one WAN in the Multilink Group is active in this state, the Multilink Up event can never happen since it is already up.

B.1.1 State Definitions

STBY_NO_ALARM: A bridge in the STBY_NO_ALARM state will report a Live/Standby Status of **Standby** and an Alarm Status of **No Alarm**. The Live/Standby Poll (Live Signal) will be generated on pin 8 and the Alarms Poll (Healthy Signal) will be generated on pin 6 of the maintenance port connector. When a bridge in the STBY_NO_ALARM state receives a Multilink Up event then the LIVE_NO_ALARM state will be entered. When a bridge in the STBY_NO_ALARM state receives a Switch Trigger event then the STBY_ALARM state will be entered. All other events will not cause a state change, status change, or signal change.

LIVE_NO_ALARM: A bridge in the LIVE_NO_ALARM state will report a Live/Standby Status of **Live** and an Alarm Status of **No Alarm**. The Live/Standby Poll (Live Signal) will be generated on pin 8 and the Alarms Poll (Healthy Signal) will be generated on pin 6 of the maintenance port connector. When a bridge in the LIVE_NO_ALARM state receives an Alarm Trigger event then the LIVE_ALARM state will be entered. When a bridge in the LIVE_NO_ALARM state receives a Multilink Down event then the RDY_ALARM state will be entered. When a bridge in the LIVE_NO_ALARM state receives a Switch Trigger then the STBY_ALARM state will be entered. All other events will not cause a state change, status change, or signal change.

Only one LAPB link needs to be up to enter the LIVE_NO_ALARM state. But, if after 10 seconds, the remaining Multilink Group member do not come up, then the Alarm Trigger event will cause the bridge to enter the LIVE_ALARM state.

LIVE_ALARM: A bridge in the LIVE_ALARM state will report a Live/Standby Status of **Live** and an Alarm Status of **Alarm**. The Live/Standby Poll (Live Signal) will be generated on pin 8 and the Alarms Poll (Healthy Signal) will cease on pin 6 of the maintenance port connector. When a bridge in the LIVE_ALARM state receives a Clear Alarm event then the LIVE_NO_ALARM state will be entered. When a bridge in the LIVE_ALARM state receives a Multilink Down event then the RDY_ALARM state will be entered. When a bridge in the LIVE_ALARM state receives a Switch Trigger event then the STBY_ALARM state will be entered. All other events will not cause a state change, status change, or signal change.

STBY_ALARM: A bridge in the STBY_ALARM state will report a Live/Standby Status of **Standby** and an Alarm Status of **Alarm**. The Live/Standby Poll (Live Signal) will cease on pin 8 and the Alarms Poll (Healthy Signal) will cease on pin 6 of the maintenance port

connector. When a bridge in the STBY_ALARM state receives a Clear Alarm event then the STBY_NO_ALARM state will be entered. All other events will not cause a state change, status change, or signal change.

RDY_ALARM: A bridge in the RDY_ALARM state will report a Live/Standby Status of **Standby** and an Alarm Status of **Alarm**. The Live/Standby Poll (Live Signal) will be generated on pin 8 and the Alarms Poll (Healthy Signal) will cease on pin 6 of the maintenance port connector. When a bridge in the RDY_ALARM state receives a Clear Alarm event then the STBY_NO_ALARM state will be entered. When a bridge in the RDY_ALARM state receives a Multilink Up event then the LIVE_ALARM state will be entered. All other events will not cause a state change, status change, or signal change.

The RDY_ALARM state is very similar to the STBY_ALARM state except that the Live/Standby Poll (Live Signal) will be generated. This state is entered when all members of the Multilink Group have been down for more than 15 seconds, or when the LAN frames missed and deleted count causes an Alarm Trigger event when in the STBY_NO_ALARM state.

B.1.2 Event Definitions

Clear Alarm: The Clear Alarm event is generated by the CLEAR MONITOR ALARM operator command, or by writing the value of 2 in the `accReutersClearAlarm` MIB variable, or via the Clear command while utilizing the Continuous Status Monitor utility.

Multilink Up: The Multilink Up event is generated by the Multilink code after at least one LAPB interface in the Multilink Group indicates Link Up and the Multilink protocol is established.

Multilink Down: The Multilink Down event is generated by the Multilink code after all LAPB interfaces in the Multilink Group indicate Link Down and remains down for 15 seconds.

Alarm Trigger: The Alarm Trigger event is generated by the detection of the following failures:

1. If a LAPB Link is Up, then the Alarm is triggered if it receives a link down indication.
2. If a LAPB Link is Down, then the Alarm is triggered if it doesn't receive a Link Up within 10 seconds after the Multilink Up event. This timer will not be re-started until

the next Multilink Up event. Clearing the alarm will not re-start the timer.

3. WAN Frame reception errors threshold exceeded (bad FCS, non-octet aligned).
4. Multilink frames lost threshold exceeded (unavailable resources, exceeding queues).
5. LAN frames missed and deleted threshold exceeded (CRC, framing, overflow).
6. CPU Load Average threshold exceeded.

Switch Trigger: The Switch Trigger event is generated by the detection of the following failures:

1. LAN transmit failure threshold exceeded (loss of carrier, late collisions, retries exceeded).
2. LAN receive interpacket time exceeded.
3. SET BRIDGE STANDBY operator command.