



# Carrier Voice over IP Performance Management Operational Measurements Volume 3

## ATTENTION

The Carrier VoIP Performance Management Operation Measurements Reference document uses four volumes to describe operational measurements (listed alphabetically) that provide information on how to load various components of the DMS switch.

## New in this release

The following new OMs have been added to this volume for release SN09U:

- NCAS\_LINK
- NMSH323
- NMSNCAS
- PKTMA

## Introduction

Operational Measurements (OMs) provide information on how to load various Components of the DMS switch. Periodic scans of switch parts and activities allow you to gather OM information. Specified parameters

define the collection, storage, and transmission of data. OMs provide the following types of data:

- Description
- Registers
  - event counts. Peg counts are registers that increase each time an event occurs.
  - usage counts. Usage counts scan or sample equipment at equal intervals. These counts increase registers when the scan detects equipment in a specific state.
- Associated OM groups

OM information can appear on a terminal or printer. The system can transmit the information to a recording device for additional processing. To request data display at a specified output device, you can enter user commands. You can enter data in tables to schedule the output of the data in advance.

For more information on how to set up an OM system, refer to the *DMS-100 Family Basic Administration Procedures*, 297-1001-300.

## In this document

The OMs in an office are dynamic and depend on the switch type(s). For comprehensiveness, this document describes OMs available in an office type of OFFCOMB. The document also describes OMs that are associated with specific components and OMs that are common in Carrier VoIP and DMS.

The following table lists the OM groups alphabetically (from N to P) and whether they are associated with a component or common in Carrier VoIP and DMS. For a description of an OM group, click on the OM group name.

### OM groups available in Carrier VoIP (Sheet 1 of 7)

| Name                     | Description                                   | Device, Manager or Application |
|--------------------------|-----------------------------------------------|--------------------------------|
| <a href="#">NACDGRP1</a> | Networked Automatic Call Distribution Group 1 | Common                         |
| <a href="#">NACDGRP2</a> | Networked Automatic Call Distribution Group 2 | Common                         |
| <a href="#">NARUSAGE</a> | Network Access Registers Usage                | Common                         |

**OM groups available in Carrier VoIP (Sheet 2 of 7)**

| <b>Name</b>                      | <b>Description</b>                                         | <b>Device, Manager or Application</b> |
|----------------------------------|------------------------------------------------------------|---------------------------------------|
| <a href="#"><u>NCAS_LINK</u></a> | NCAS LINK                                                  | Common                                |
| <a href="#"><u>NCASLINK</u></a>  | NCAS link                                                  | MCS 5200                              |
| <a href="#"><u>NCMCPUST</u></a>  | Non-computing Module Node Central Processing Unit Status   | Common                                |
| <a href="#"><u>NDS0CARR</u></a>  | 8-port 64-Kbps Non-multiplexed Digital Voice/data Carriers | Common                                |
| <a href="#"><u>NETMSG</u></a>    | Network Message Service                                    | Common                                |
| <a href="#"><u>NIUFBUS</u></a>   | Network Interface Unit (NIU) Frame Transport Bus (F-bus)   | Common                                |
| <a href="#"><u>NIUMEMUT</u></a>  | Network Interface Unit Memory Use                          | Common                                |
| <a href="#"><u>NMC</u></a>       | Network Module Controller Maintenance Summary              | Common                                |
| <a href="#"><u>NMSNCAS</u></a>   | NMS messages over NCAS link                                | Common                                |
| <a href="#"><u>NMTCLINK</u></a>  | Node Maintenance - Link                                    | MG4K                                  |
| <a href="#"><u>NMTCNODE</u></a>  | Node Maintenance - Node                                    | MG4K                                  |
| <a href="#"><u>NMTCTYPE</u></a>  | Node Maintenance - Node Type                               | MG4K                                  |
| <a href="#"><u>NMTCUNIT</u></a>  | Node Maintenance - Unit                                    | IW_SPM_IP                             |
| <a href="#"><u>NPAPEG</u></a>    | Numbering Plan Area                                        | Common                                |
| <a href="#"><u>NRS</u></a>       | Network Resource Selector                                  | Common                                |
| <a href="#"><u>NSC</u></a>       | Number Services Code Call Summary                          | Common                                |
| <a href="#"><u>NSCACG</u></a>    | Number Services Code Automatic Call Gapping                | Common                                |
| <a href="#"><u>NWMFRRCT</u></a>  | Network Management Flexible Reroute                        | Common                                |
| <a href="#"><u>NWMSILC</u></a>   | Network Management Selective Incoming Load Control         | Common                                |

**OM groups available in Carrier VoIP (Sheet 3 of 7)**

| <b>Name</b>                      | <b>Description</b>                                                                 | <b>Device, Manager or Application</b> |
|----------------------------------|------------------------------------------------------------------------------------|---------------------------------------|
| <a href="#"><u>NWMTGCNT</u></a>  | Network Management Trunk Group Control                                             | Common                                |
| <a href="#"><u>NX25L2</u></a>    | NX25 Level 2                                                                       | Common                                |
| <a href="#"><u>NX25L3</u></a>    | NX25 Level 3                                                                       | Common                                |
| <a href="#"><u>NX25MLP</u></a>   | NX25 Multilink Procedures                                                          | Common                                |
| <a href="#"><u>OADATCOM</u></a>  | Operator Services System Advanced Intelligent Network (OSSAIN) Data Communications | Common                                |
| <a href="#"><u>OAFLTRIG</u></a>  | OSSAIN Float Triggers                                                              | Common                                |
| <a href="#"><u>OAINNODE</u></a>  | OSSAIN Node Maintenance                                                            | Common                                |
| <a href="#"><u>OAINQMS</u></a>   | OSSAIN Queue Management System                                                     | Common                                |
| <a href="#"><u>OAINRTE</u></a>   | OSSAIN Route                                                                       | Common                                |
| <a href="#"><u>OANODEDC</u></a>  | OSSAIN Node Data Communications                                                    | Common                                |
| <a href="#"><u>OAPCALP1</u></a>  | Open Automated Protocol (OAP) Call Processing 1                                    | Common                                |
| <a href="#"><u>OAPCALP2</u></a>  | Open Automated Protocol (OAP) Call Processing 2                                    | Common                                |
| <a href="#"><u>OAPCALP3</u></a>  | OAP Call Processing 3                                                              | Common                                |
| <a href="#"><u>OAPCALP4</u></a>  | OAP Call Processing 4                                                              | Common                                |
| <a href="#"><u>OAPCALP5</u></a>  | OAP Call Processing 5                                                              | Common                                |
| <a href="#"><u>OAPCALP6</u></a>  | OAP Call Processing 6                                                              | Common                                |
| <a href="#"><u>OAPCALP7</u></a>  | OAP Call Processing 7                                                              | Common                                |
| <a href="#"><u>OAPCALP8</u></a>  | OAP Call Processing 8                                                              | Common                                |
| <a href="#"><u>OAPCALP9</u></a>  | OAP Call Processing 9                                                              | Common                                |
| <a href="#"><u>OAPCALP10</u></a> | OAP Call Processing 10                                                             | Common                                |



**OM groups available in Carrier VoIP (Sheet 4 of 7)**

| <b>Name</b>                     | <b>Description</b>                                    | <b>Device, Manager or Application</b> |
|---------------------------------|-------------------------------------------------------|---------------------------------------|
| <a href="#"><u>OAPMERRN</u></a> | OAP Message Error - Node                              | Common                                |
| <a href="#"><u>OAPMERRS</u></a> | OAP Message Error - Session Pool                      | Common                                |
| <a href="#"><u>OAPMTYPN</u></a> | OAP Message Type - Node                               | Common                                |
| <a href="#"><u>OAPMTYPS</u></a> | OAP Message Type - Session Pool                       | Common                                |
| <a href="#"><u>OAPNMIS</u></a>  | OAP node management information system                | Common                                |
| <a href="#"><u>OAPNMTC</u></a>  | OAP Node Maintenance Operations and Responses         | Common                                |
| <a href="#"><u>OAPSPMTC</u></a> | OAP Session Pool Maintenance Operations and Responses | Common                                |
| <a href="#"><u>OASNPLDC</u></a> | OSSAIN Session Pool Data Communications               | Common                                |
| <a href="#"><u>OASNPOOL</u></a> | Session Pool Inventory                                | Common                                |
| <a href="#"><u>OASVNDCP</u></a> | OSSAIN Service Node Call Processing                   | Common                                |
| <a href="#"><u>OFF250</u></a>   | Offhook 250                                           | Common                                |
| <a href="#"><u>OFZ</u></a>      | Office Traffic Summary                                | Common                                |
| <a href="#"><u>OFZ2</u></a>     | Office Traffic Extension Summary                      | Common                                |
| <a href="#"><u>OFZ2NET1</u></a> | Office to Network Group 1                             | CS2K                                  |
| <a href="#"><u>OFZ2NET2</u></a> | Office to Network Group 2                             | CS2K                                  |
| <a href="#"><u>OGTQMS</u></a>   | Outgoing trunk queue management system                | Common                                |
| <a href="#"><u>OHBTDTU</u></a>  | Off-Hook Balance Test Digital Test Unit               | Common                                |
| <a href="#"><u>OHBTRES</u></a>  | Off-hook balance testing                              | Common                                |
| <a href="#"><u>OHBTTYPE</u></a> | Off-Hook Balance Test Results per test type           | Common                                |

**OM groups available in Carrier VoIP (Sheet 5 of 7)**

| <b>Name</b>                                   | <b>Description</b>                                             | <b>Device, Manager or Application</b> |
|-----------------------------------------------|----------------------------------------------------------------|---------------------------------------|
| <a href="#"><u>OHQCBQCG</u></a>               | Off-hook queuing and call back queuing per customer group      | Common                                |
| <a href="#"><u>OHQCBQR2</u></a>               | Off-hook queuing and call back queuing for table IBNRT2 routes | Common                                |
| <a href="#"><u>OHQCBQR3</u></a>               | Off-hook queuing and call back queuing for table IBNRT3 routes | Common                                |
| <a href="#"><u>OHQCBQR4</u></a>               | Off-hook queuing and call back queuing for table IBNRT4 routes | Common                                |
| <a href="#"><u>OHQCBQRT</u></a>               | Off-hook queuing and call back queuing per route               | Common                                |
| <a href="#"><u>OPCHOICE</u></a>               | Operator Choice                                                | Common                                |
| <a href="#"><u>OracleRepLinkStats</u></a>     | Oracle Replication Link Statistics                             | MCS 5200                              |
| <a href="#"><u>OracleSysTableCounters</u></a> | Oracle System Table Counters                                   | MCS 5200                              |
| <a href="#"><u>ONI</u></a>                    | Operator number identification                                 | Common                                |
| <a href="#"><u>OSACCP1</u></a>                | OSAC Call Processing 1                                         | Common                                |
| <a href="#"><u>OSACCP2</u></a>                | OSAC Call Processing 2                                         | Common                                |
| <a href="#"><u>OSACND</u></a>                 | OSAC Node Maintenance                                          | Common                                |
| <a href="#"><u>OSACSP</u></a>                 | OSAC Session Pool Maintenance                                  | Common                                |
| <a href="#"><u>OSNND</u></a>                  | OSN Node                                                       | Common                                |
| <a href="#"><u>OSNSP</u></a>                  | OSN Session Pool                                               | Common                                |
| <a href="#"><u>OTS</u></a>                    | Office Traffic Summary                                         | Common                                |
| <a href="#"><u>PCMCARR</u></a>                | CCITT DS30 digital carrier maintenance summary                 | Common                                |
| <a href="#"><u>PCNF</u></a>                   | Preset conference                                              | Common                                |
| <a href="#"><u>PJM</u></a>                    | Persistence Job Manage                                         | MCS 5200                              |

**OM groups available in Carrier VoIP (Sheet 6 of 7)**

| <b>Name</b>                           | <b>Description</b>                                                                               | <b>Device, Manager or Application</b> |
|---------------------------------------|--------------------------------------------------------------------------------------------------|---------------------------------------|
| <a href="#">PKTMA</a>                 | Packet Media Anchor                                                                              | Common                                |
| <a href="#">PM</a>                    | Peripheral Module                                                                                | IW_SPM_IP                             |
| <a href="#">PM1</a>                   | Peripheral module single-unit maintenance summary                                                | Common                                |
| <a href="#">PM2</a>                   | Dual-unit peripheral module maintenance summary                                                  | Common                                |
| <a href="#">PMMSGCNT</a>              | Peripheral module message counter                                                                | Common                                |
| <a href="#">PMOVL</a>                 | Peripheral module overloaded                                                                     | Common                                |
| <a href="#">PMSTAT</a>                | Peripheral module status                                                                         | Common                                |
| <a href="#">PMTYP</a>                 | Peripheral Module Type                                                                           | IW_SPM_IP                             |
| <a href="#">PPCO</a>                  | Pre Paid Coin Overtime                                                                           | Common                                |
| <a href="#">PRADCHL2</a>              | PRA D-channel layer 2 performance summary                                                        | Common                                |
| <a href="#">PRAFAC</a>                | Primary rate access facility                                                                     | Common                                |
| <a href="#">PRASERV</a>               | Primary Rate Access Service                                                                      | Common                                |
| <a href="#">Presence</a>              | Presence                                                                                         | MCS 5200                              |
| <a href="#">Presence_Event_Report</a> | Presence Event Report                                                                            | MCS 5200                              |
| <a href="#">PRIMWIC</a>               | Primary rate interface (PRI) access interface that has a Message Waiting Indicator (MWI) Control | Common                                |
| <a href="#">PRISVCS</a>               | PRI services                                                                                     | Common                                |
| <a href="#">PRKOM</a>                 | Call park operational measurement                                                                | Common                                |
| <a href="#">PRP</a>                   | Preroute peg                                                                                     | Common                                |
| <a href="#">PSN_ERDC</a>              | Programmable Service Node (PSN) Error — Data Communication Level                                 | Common                                |

**OM groups available in Carrier VoIP (Sheet 7 of 7)**

| <b>Name</b>              | <b>Description</b>                | <b>Device, Manager or Application</b> |
|--------------------------|-----------------------------------|---------------------------------------|
| <a href="#">PSN_ERFM</a> | PSN Error in Finite State Machine | Common                                |
| <a href="#">PSN_ERPS</a> | PSN Primitive Processing Error    | Common                                |
| <a href="#">PSN_FCTR</a> | PSN Flow Control                  | Common                                |
| <a href="#">PSN_NOTF</a> | PSN Notifications Sent            | Common                                |
| <a href="#">PSN_PRIM</a> | Programmable Service Node         | Common                                |
| <a href="#">PSN_USAG</a> | PSN Usage                         | Common                                |
| <a href="#">Publish</a>  | Publish                           | MCS 5200                              |

**Supplementary OMs**

The following documents reference OMs that do not appear in this document:

- *North American DMS-100 Operations Measurements Reference Manual*, 297-8021-814
- *Carrier VoIP SN07 OSS (ATM and IP) Advance Feature Guide*, PLN-07AT-OSS
- *Carrier VoIP Fault Management Logs Reference*, NN10275-909

## NACDGRP1

### Description

OM group Networked Automatic Call Distribution Group 1 (NACDGRP1) records the total ACD traffic for the NACD groups. These registers record calls that overflow from or to a NACD group because of immediate overflow or time delay overflow.

The following table lists the key and info fields associated with OM group NACDGRP1:

| Key field     | Info field |
|---------------|------------|
| NACD_OM_INDEX | None       |

### Related functional groups

There are no functional groups associated with OM group NACDGRP1.

### Registers

The following table lists the registers associated with OM group NACDGRP1 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NACDGRP1 (Sheet 1 of 2)

| Register name            | Measures                                            |
|--------------------------|-----------------------------------------------------|
| <a href="#">IMINFLCL</a> | Immediate inflowed from a local group               |
| <a href="#">IMINFQED</a> | Immediate inflowed calls queued                     |
| <a href="#">IMINFREM</a> | Immediate inflowed from a remote group              |
| <a href="#">IMMTMOFL</a> | Immediate inflowed calls overflowed                 |
| <a href="#">IMOFLLCL</a> | Immediate overflows to a local group                |
| <a href="#">IMOFLREM</a> | Immediate overflows to a remote group               |
| <a href="#">LOGQFULL</a> | Logical queue is full                               |
| <a href="#">LOGQLCL</a>  | Logically queued caused by request from local group |
| <a href="#">NOOFLGRP</a> | No overflow group                                   |

**Registers for OM group NACDGRP1 (Sheet 2 of 2)**

| Register name            | Measures                                                     |
|--------------------------|--------------------------------------------------------------|
| <a href="#">PHYQLOGQ</a> | Physically queued and also logically queued at a local group |
| <a href="#">TFAILLCL</a> | Time overflow to a local group fails                         |
| <a href="#">TFAILREM</a> | Time overflow to a remote group fails                        |
| <a href="#">TMANSLCL</a> | Time overflow call answered by the local source group        |
| <a href="#">TMANSREM</a> | Time overflow call answered by the remote source group       |
| <a href="#">TMINFLCL</a> | Time inflowed from a local group                             |
| <a href="#">TMINFREM</a> | Time inflowed from a remote group                            |
| <a href="#">TMOFLCL</a>  | Time overflow to a local group                               |
| <a href="#">TMOFLREM</a> | Time overflow to a remote group                              |
| <a href="#">USRABNDN</a> | User abandons while the call is logically queued             |

**IMINFLCL****Register type**

Peg

**Description**

IMINFLCL increases when a call arrives at an NACD group because of the immediate overflow from a local NACD group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**IMINFQED****Register type**

Peg

**Description**

IMINFQED increases when the system queues a call that arrived at this NACD group because of immediate overflow from a local or remote source group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**IMINFREM****Register type**

Peg

**Description**

IMINFREM increases when a call arrives at an NACD group because of immediate overflow from a remote NACD group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**IMMTMOFL****Register type**

Peg

**Description**

IMMTMOFL increases each time a call arrives at this NACD group because of immediate overflow. The NACD group tries to overflow the call to a local or remote overflow group. The call queues for a period of time longer than the time delay overflow time.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**IMOFLCL****Register type**

Peg

**Description**

IMOFLCL increases when an NACD group overflows an incoming call to a local overflow group because of exceeded queue or wait thresholds.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**IMOFLREM****Register type**

Peg

**Description**

IMOFLREM increases when an attempt to overflow an incoming call to a remote overflow group occurs. The register increases when the system marks this attempt because of exceeded queue or wait thresholds.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**LOGQFULL****Register type**

Peg



**Description**

LOGQFULL increases when a call fails to queue logically because the logical queue is full.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**LOGQLCL****Register type**

Peg

**Description**

LOGQLCL increases when a queued call is at this group. A call is logically queued while it remains queued at a local source NACD group.

**Associated registers**

ACDGRP\_ACDTMINF, which increments each time a call is logically queued at this NACD group

**Extension registers**

None

**Associated logs**

None

**NOOFLGRP****Register type**

Peg

**Description**

NOOFLGRP increases when the system cannot find a best overflow group to overflow:

- a new incoming call
- a queued call that waited over the time delay overflow time

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PHYQLOGQ****Register type**

Peg

**Description**

PHYQLOGQ increases when a call is physically queued at this group and logically queued at a local overflow group.

**Associated registers**

ACDGRP\_ACDTMOFL, which increases each time a call is time overflowed from this NACD group to another NACD group

**Extension registers**

None

**Associated logs**

None

**TFAILLLCL****Register type**

Peg

**Description**

TFAILLLCL increases when an attempt to time overflow from this NACD group to a local overflow group fails. Register TFAILLLCL increases for one of the following reasons:

- the overflow group is in Night Service or has controlled interflow (CIF) active
- all agents in the overflow group are in make set busy (MSB) mode
- the call cannot be queued logically because the logical queue exceeds the logical queue size or is set to zero
- the caller abandons the call
- the group of the call answers the call

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**TFAILREM****Register type**

Peg

**Description**

TFAILREM increases when an attempt to time overflow a call from this NACD group (source group) to a remote overflow group fails. This register TFAILREM increases when the attempt fails for one of the following reasons:

- this NACD group cannot send an NACD Reroute Request message
- no response was received in the TCAP T1 (NCAD resend timer) duration after this NACD group cannot resends an NACD Reroute Request message
- the overflow group rejects an NACD Reroute Request message, the overflow group or the source group cancels the message

**Associated registers**

The following registers are associated with TFAILREM:

- NACDGRP2\_CANCRECD, which increases when an NACD group receives an NACD Cancel Request message
- NACDGRP2\_CANCSENT, which increases when an NACD group sends an NACD Cancel Request message to cancel an NACD Reroute Request message
- NACDGRP2\_REJRECD, which increases when an NACD group receives an NACD Reject Request message.
- NACDGRP2\_QRYSENT, which increases when an NACD group sends out an NACD Reroute Request message.
- [TMOFLREM](#)

**Validation formulas**

The following formulas relate to TRAILREM and its associated registers:

- $\text{NACDGRP2\_QRYSENT} = \text{TMOFLREM} + \text{TFAILREM}$
- $\text{TFAILREM} = \text{NACDGRP2\_CANCSENT} + \text{NACDGRP2\_CANCRECD} + \text{NACDGRP2\_REJRECD}$

**Extension registers**

None

**Associated logs**

None

**TMANSLCL****Register type**

Peg

**Description**

TMANSLCL increases when the local source NACD group answers a logically queued call.

**Associated registers**

The following registers are associated with TMANSLCL:

- ACDGRP\_ACDTMANS, which increases each time another NACD group answers a logically queued call at this NACD group
- ACDGRP\_ACDTMOFL, which increases when this NACD group time overflows to another NACD group.
- [TMANSLCL](#)

**Validation formula**

$ACDGRP\_ACDTMANS = TMANSLCL + TMANSREM + ACDGRP\_ACDTMOFL$

**Extension registers**

None

**Associated logs**

None

**TMANSREM****Register type**

Peg

**Description**

TMANSREM increases when a remote source group answers a queued call logically at this NACD group.

**Note:** NACDGRP2\_CANCRECD increases as well.

**Associated registers**

The following registers are associated with TMANSREM:

- ACDGRP\_ACDTMANS, which increases when another NACD group answers a logically overflowed call at this NACD group
- ACDGRP\_ACDTMOFL, which increases when this NACD time overflows a call to another NACD group
- [TMANSLCL](#)

**Validation formula**

$ACDGRP\_ACDTMANS = TMANSLCL + TMANSREM + ACDGRP\_ACDTMOFL$

**Extension registers**

None

**Associated logs**

None

**TMINFLCL****Register type**

Peg

**Description**

TMINFLCL increases when a call arrives at an NACD group because of time delay overflow from a local NACD group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

MS303

**TMINFREM****Register type**

Peg

**Description**

TMINFREM increases when a call arrives at an NACD group. A call arrives at an NACD group because of a time delay overflow from a remote NACD group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**TMOFLLCL****Register type**

Peg

**Description**

TMOFLLCL increases when a queued call at this NACD group is time overflowed to a local overflow group. The register increases if the call overflows because the call waits in the queue over the time delay overflow time.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**TMOFLREM****Register type**

Peg

**Description**

TMOFLREM increases when a queued call is time overflowed to a remote overflow group. The queued call is at the associated NACD group. A queued call is overflowed because the queued call waits in the call queue longer than the time delay overflow time.

**Associated registers**

The following registers are associated with TMOFLREM:

- NACDGRP2\_QRYSENT, which increases each time an NACD Reroute Request message is sent out from an NACD group
- [TFAILREM](#)

**Validation formula**
$$\text{NACDGRP2\_QRYSENT} = \text{TMOFLREM} + \text{TFAILREM}$$

**Extension registers**

None

**Associated logs**

None

**USRABNDN****Register type**

Peg

**Description**

USRABNDN increases when the caller (user) abandons a logically queued call at this group.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NACDGRP2

### Description

OM group Networked Automatic Call Distribution Group 2 (NACDGRP2) provides transaction capabilities application part (TCAP) message counts specific to Network Automatic Call Distribution (NACD) applications.

The following table lists the key and info fields associated with OM group NACDGRP2:

| Key field     | Info field |
|---------------|------------|
| NACD_OM_INDEX | None       |

### Related functional groups

There are no functional groups associated with OM group NACDGRP2.

### Registers

The following table lists the registers associated with OM group NACDGRP2 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NACDGRP2 (Sheet 1 of 2)

| Register name             | Measures                         |
|---------------------------|----------------------------------|
| <a href="#">ACKRECD</a>   | Acknowledge received             |
| <a href="#">ACKSENT</a>   | Acknowledge sent                 |
| <a href="#">CANCRECD</a>  | Cancel message received          |
| <a href="#">CANCSENT</a>  | Cancel message sent              |
| <a href="#">CCREPLYR</a>  | Reply to cancel message received |
| <a href="#">CCREPLYS</a>  | Reply to cancel message sent     |
| <a href="#">FREERECD</a>  | Free agent message received      |
| <a href="#">FREESSENT</a> | Free agent message sent          |
| <a href="#">QRYRECD</a>   | Query received                   |



**Registers for OM group NACDGRP2 (Sheet 2 of 2)**

| Register name            | Measures                           |
|--------------------------|------------------------------------|
| <a href="#">QRYSENT</a>  | Query sent                         |
| <a href="#">REJRECD</a>  | Reject message received            |
| <a href="#">REJSENT</a>  | Rejection message sent             |
| <a href="#">RESENDTO</a> | Resend timer timeout               |
| <a href="#">RESRVDTO</a> | Reservation timer timeout          |
| <a href="#">RIRECD</a>   | Resource index (RI) received       |
| <a href="#">RIREPLYR</a> | Resource index (RI) reply received |
| <a href="#">RIREPLYS</a> | Resource index (RI) reply sent     |
| <a href="#">RISENT</a>   | Resource index (RI) sent           |
| <a href="#">SCREPLYR</a> | Service complete reply received    |
| <a href="#">SCREPLYS</a> | Service complete reply sent        |
| <a href="#">SRVCOMPR</a> | Service complete message received  |
| <a href="#">SRVCOMPS</a> | Service complete message sent      |
| <a href="#">T4TMEOUT</a> | Total TCAP outgoing messages       |
| <a href="#">TTCAPIN</a>  | Total TCAP incoming messages       |
| <a href="#">TTCAPOUT</a> | Total TCAP outgoing messages       |

**ACKRECD****Register type**

Peg

**Description**

ACKRECD increases when an NACD group receives an NACD Acknowledge Request message in response to an NACD Reroute Request message.

**Associated registers**

The following registers are associated with ACKRECD:

- ACDGRP\_ACDTMOFL, which increases when this NACD time overflows to another NACD group
- NACDGRP1\_PHYQLOGQ, which increases when a call is physically queued at this group, and logically queued at a local overflow group

**Validation formula**
$$\text{ACDGRP\_ACDTMOFL} = \text{NACDGRP1\_PHYQLOGQ} + \text{ACKRECD}$$
**Extension registers**

None

**Associated logs**

None

**ACKSENT****Register type**

Peg

**Description**

ACKSENT increases when the NACD group sends an NACD Acknowledge Request message in response to an NACD Reroute Request message. The NACD Acknowledge Request message indicates that a call is logically queued at this NACD group.

**Associated registers**

The following registers are associated with ACKSENT:

- ACDGRP\_ACDTMINF, which increases when a call is logically queued at this NACD group
- NACDGRP1\_LOGQLCL, which increases when a call is logically queued at this group while the call remains queued at a local source NACD group

**Validation formula**
$$\text{ACDGRP\_ACDTMINF} = \text{NACDGRP1\_LOGQLCL} + \text{ACKSENT}$$
**Extension registers**

None

**Associated logs**

None

**CANCRECD****Register type**

Peg

**Description**

CANCRECD increases when an NACD group receives a Cancel Request message.

**Associated registers**

The following registers are associated with CANCRECD:

- [CANSENT](#), [REJRECD](#)
- TFAILREM, which increases when an attempt to time overflow a call from this NACD source group to a remote overflow group fails

**Validation formula**
$$\text{TFAILREM} = \text{CANSENT} + \text{CANCRECD} + \text{REJRECD}$$
**Extension registers**

None

**Associated logs**

None

**CANSENT****Register type**

Peg

**Description**

CANSENT increases when an NACD group sends a Cancel Request message to cancel an NACD Reroute Request message.

The source group cancels an NACD Reroute Request message when a caller abandons a time overflow. The source group also cancels an NACD Reroute message when the original source group answers a time overflow. The overflow group cancels an NACD Reroute Request message when either the T4 duration timer or the T2 reservation timer expires.

**Associated registers**

The following registers are associated with CANSENT:

- [CANCRECD](#), [REJRECD](#)
- TFAILREM, which increases for each failed attempt to time overflow a call from this NACD source group to a remote overflow group

**Validation formula**

$\text{TFAILREM} = \text{CANCSNT} + \text{CANCRECD} + \text{REJRECD}$

**Extension registers**

None

**Associated logs**

None

**CCREPLYR****Register type**

Peg

**Description**

CCREPLYR increases when an NACD group receives a Cancel Request reply.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**CCREPLYS****Register type**

Peg

**Description**

CCREPLYS increases when an NACD group sends a Cancel Request reply.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**FREERECD****Register type**

Peg

**Description**

FREEECD increases when this register receives an NACD Agent Free message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**FREESENT****Register type**

Peg

**Description**

FREESENT increases when an NACD group sends an NACD Free Agent message to indicate that the group reserved a free agent.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**QRYRECD****Register type**

Peg

**Description**

QRYRECD increases when an NACD group receives an NACD Reroute Request message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**QRYSENT****Register type**

Peg

**Description**

QRYSENT increases when an NACD group sends an NACD Reroute Request message.

**Associated registers**

The following registers are associated with QRYSENT:

- NACDGRP1\_TFAILREM, which increases when a call time overflow from this NACD group (source group) to a remote overflow group fails
- NACDGRP1\_TMOFLREM, which increases when a queued call at this NACD group time overflows to a remote overflow group. The register increases when the queued call overflows because the call waited over the time delay overflow time.

**Validation formula**
$$\text{QRYSENT} = \text{NACDGRP1\_TMOFLREM} + \text{NACDGRP1\_TFAILREM}$$
**Extension registers**

None

**Associated logs**

None

**REJRECD****Register type**

Peg

**Description**

REJRECD increases when an NACD group receives an NACD Reject Request message.

**Associated registers**

The following registers are associated with CANCSENT:

- [CANCRECD](#), [CANCSENT](#)
- TFAILREM, which increases for each failed attempt to time overflow a call from this NACD source group to a remote overflow group.

**Validation formula**
$$\text{TFAILREM} = \text{CANCSENT} + \text{CANCRECD} + \text{REJRECD}$$

**Extension registers**

None

**Associated logs**

None

**REJSENT****Register type**

Peg

**Description**

REJSENT increases when an NACD group sends an NACD Reject Request message to reject an NACD Reroute Request. An NACD group sends the reject message for one of the following reasons:

- the request message has protocol errors or contains invalid information
- the overflow group is in Night Service
- the logical queue of the overflow group is full
- no software resource is available

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RESENDTO****Register type**

Peg

**Description**

RESENDTO increases when an NACD Reroute Request message is resent but the TCAP resend timer (T1) duration does not receive a response.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RESRVDTO****Register type**

Peg

**Description**

RESRVDTO increases when a reservation timer (T2) expires. The T2 timer starts when a free agent reserves after an NACD Reroute Request message receives.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RIRECD****Register type**

Peg

**Description**

RIRECD increases when this register receives an NACD Status Update or Status Exchange message that contains an RI for an NACD group.

**Associated registers**

None

**Extension registers**

RIRECD2

**Associated logs**

None

**RIREPLYR****Register type**

Peg

**Description**

RIREPLYR increases when an NACD group receives an NACD Status Reply message that contains an RI for a NACD group.



**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RIREPLYS****Register type**

Peg

**Description**

RIREPLYS increases when this register sends an NACD Status Reply message to return the RI. The group that sent an NACD Status Exchange message receives the RI.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RISENT****Register type**

Peg

**Description**

RISENT increases when an NACD group sends an NACD Status Update or Status Exchange message. An NACD group sends a message to broadcast the RI to the remote network groups of the group.

**Associated registers**

None

**Extension registers**

RISENT2

**Associated logs**

None

**SCREPLYR****Register type**

Peg

**Description**

SCREPLYR increases when an NACD group receives an NACS Service Complete Reply message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SCREPLYS****Register type**

Peg

**Description**

SCREPLYS increases when an NACD group sends an NACD Service Complete Reply message to acknowledge arrival of the NACD Service Complete message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SRVCOMPR****Register type**

Peg

**Description**

SRVCOMPR increases when an NACD group receives an NACD Service Complete message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SRVCOMPS****Register type**

Peg

**Description**

SRVCOMPS increases when an NACD group sends an NACD Service Complete message for the following reasons:

- to show that a call is time-overflowed to a remote group
- to show that a TCAP transaction can be closed

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**T4TMEOUT****Register type**

Peg

**Description**

T4TMEOUT increases when a TCAP T4 timer expires. The T4 timer starts when a call is logically queued or a free agent is reserved. The T4 timer starts when the switch receives an NACD Reroute Request message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**TTCAPIN****Register type**

Peg

**Description**

TTCAPIN increases when an NACD group receives an NACD TCAP message. The TTCAPIN records the number of incoming TCAP messages an NACD group receives.

**Associated registers**

None

**Extension registers**

TTCAPIN2

**Associated logs**

None

**TTCAPOUT****Register type**

Peg

**Description**

TTCAPOUT increases when an NACD group sends an NACD TCAP message. Register TTCAPOUT records the number of outgoing TCAP messages that originate from an NACD group.

**Associated registers**

None

**Extension registers**

TTCAPOU2

**Associated logs**

None

---

## NARUSAGE

---

### Description

OM group Network Access Registers Usage (NARUSAGE) provides information on the use of network access registers (NAR). This OM group NARUSAGE counts attempts to use NARs and counts blocked NAR attempts. This OM group also provides a traffic use count for each NAR.

The NAR feature provides a more efficient method for “throttling” calls. This method involves regulating the ability of a Meridian Digital Centrex (MDC) call to terminate. The use of virtual facilities groups for simple call throttling requires retranslation, which increases DMS processing time for each call. The NAR feature does not require retranslation, which causes DMS processing time to improve.

The NAR feature is available at the customer group level, where default values for incoming and outgoing NAR groups are assigned. Each NAR group has a size that indicates the number of allowed simultaneous calls. The system checks the size of the group when a caller places a throttled call. If the NAR group receives the maximum number of simultaneous calls, the call does not continue.

The NAR feature also enables the assignment of incoming and outgoing NAR groups to network class of service (NCOS) groups, and translation selectors.

This feature enables the system to divert calls to an alternate route that provides throttling at the routing level. The system can divert calls to an alternate route where the NAR feature is active. The system can divert a call routed to a busy trunk group to an alternate route. The NAR on the alternate route can throttle the call.

The NAR cannot throttle all calls to and from the MDC customer group. The NAR can throttle calls that cannot terminate because the NAR does not have enough NAR resources. The path a call takes through translations and routing determines if the NAR can throttle the call.

Outgoing calls that the NAR can throttle use the following NET selector network types of tables IBNXLA and XLANAME:

- Direct Outward Dial (DOD) access
- Out WATS (OWT) access
- Electronic Switching Network (ESN) access
- Private (PVT) Network access

- General (GEN) Network access
- Multi-switched Business Group (MBG) access
- Location Code (LOC) for MBG access

Outgoing calls that the NAR can throttle use the following ROUTE selector types of tables IBNXLA and XLANAME:

- Location (L)
- Common Language Location Identifier (S)
- Table (T)

The new table NARDATA defines each NAR group. An NAR group may have 0 to 2047 units and provide 0 to 2047 simultaneous accesses. Each NAR group has an overflow route. This overflow route can be to another NAR group or to an MDC customer-defined treatment. The caller only goes to the treatment if the call is an outgoing call from the MDC customer group. The overflow NAR applies to both incoming calls to the customer group and outgoing calls from the customer group.

If the call needs to access an NAR group, the system checks the NAR group for available access. If an idle NAR unit is present, the call continues as normal. If NAR units are not present, the system checks the overflow route. If the overflow route shows an NAR name, the system checks the NAR group marked for available access.

If the overflow route shows a customer-defined treatment, the system routes the call to treatment. If the call is outgoing from the customer group, the overflow route uses the customer-defined treatment entered. If the call is incoming to the customer group, the system routes the call to customer group resource overflow (CGRO) treatment.

The system limit is five consecutive overflows when attempting access. The limit is five because an NAR group can identify another NAR group as an overflow route. When the system overflows the maximum number of times, the system routes the call to an office-wide treatment. The outgoing and incoming calls route to CGRO treatment.

The OM group NARUSAGE collects data on each NAR group defined in the office. Any access attempt on an NAR unit increases the NARTOTAL register. If all NAR part are not available, the NARBLCKD register also increases.

The following table lists the key and info fields associated with OM group NARUSAGE:

| Key field                                                                                                                    | Info field                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------|
| NAR_NAME<br>(The field NAR_NAME is assigned to the NAR as defined in table NARDATA. This field ranges from 1-16 characters.) | NAR_SIZE<br>(This field NAR_SIZE is the number of units or the number of simultaneous accesses to the NAR.) |

## Related functional groups

DMS-100 switches (provisioned with Meridian Digital Centrex and the Network Access Registers [NARS] feature) are associated with OM group MTA.

**Note:** The NARS is correct for Meridian Digital Centrex (MDC), integrated services digital network (ISDN), and Residential Enhanced Services (RES) lines.

## Registers

The following table lists the registers associated with OM group NARUSAGE and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group NARUSAGE

| Register name            | Measures                            |
|--------------------------|-------------------------------------|
| <a href="#">NARTOTAL</a> | NAR total number of access attempts |
| <a href="#">NARBLCKD</a> | NAR blocked attempts                |
| <a href="#">NARTRAF</a>  | NAR traffic usage count             |

### NARTOTAL

#### Register type

Peg

#### Description

NARTOTAL increases when an attempt to access an NAR occurs.

#### Associated registers

[NARBLCKD](#)

**Extension registers**

None

**Associated logs**

None

**NARBLCKD****Register type**

Peg

**Description**

NARBLCKD increases when the system attempts to access an NAR but units are not available.

**Associated registers**[NARTOTAL](#)**Extension registers**

None

**Associated logs**

None

**NARTRAF****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NARTRAF shows the amount of traffic that uses each NAR.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None



## NCAS\_LINK

### Description

OM group NCAS\_LINK keeps a record of the state changes of the NCAS Link and the number of messages sent and received over the NCAS link.

This OM group accurately tracks the messages sent and received on the NCAS link between CS2K Core and Session Server. The OM pegging will provide information on the message traffic between CS2K Core and Session Server. The OM pegging will also count the number of times the NCAS Link has gone down and been re-established.

The OM group NCAS\_LINK in the CS2K Core is associated with the OM group NCASLINK in the MCS. The OM group NCAS\_LINK measures the performance of the NCAS link in the CS2K Core. The OM group NCASLINK measures the performance of the NCAS link in the MCS.

The following table lists the key and info fields associated with OM group NCAS\_LINK.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group NCAS LINK.

### Registers

The following table lists the registers associated with OM group NCAS\_LINK and what they measure. For a description of a register, click on the register name.

#### Registers for OM group NCAS\_LINK

| Register name                 | Measures                                    |
|-------------------------------|---------------------------------------------|
| <a href="#">NCASLINK_UP</a>   | Number of times the NCAS LINK is brought up |
| <a href="#">NCASLINK_DOWN</a> | Number of times the NCAS LINK goes down     |

**Registers for OM group NCAS\_LINK**

| Register name                 | Measures                                                   |
|-------------------------------|------------------------------------------------------------|
| <a href="#">MSG_SENT</a>      | Number of messages successfully sent over an NCAS LINK     |
| <a href="#">MSG_RCVD</a>      | Number of messages successfully received over an NCAS LINK |
| <a href="#">MSG_SEND_FAIL</a> | Number of messages failed to send over an NCAS LINK        |
| <a href="#">MSG_RCV_FAIL</a>  | Number of messages failed to receive over an NCAS LINK     |

**NCASLINK\_UP****Register type**

Peg

**Description**

*linkUp* counts the number of times the logical link is transitioned into a connected state due to administrative action, network repair, etc.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCASLINK\_DOWN****Register type**

Peg

**Description**

*linkDown* counts the number of times the logical link is transitioned into a disconnected state due to administrative action, network failure, etc.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MSG\_SENT****Register type**

Peg

**Description**

*msgSent* records the number of messages successfully sent over the link.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MSG\_RCVD****Register type**

Peg

**Description**

*msgRcvd* counts the number of messages successfully received over the link.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MSG\_SEND\_FAIL****Register type**

Peg

**Description**

*msgSendFail* counts the number of messages that failed to be sent.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

***MSG\_RCV\_FAIL*****Register type**

Peg

**Description***msgRcvrFail* counts the number of messages that failed to be received.**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NCASLINK

### Description

The OM group NCASLINK provides data regarding the state and usage of the NCAS link. Applications use this link for non-call-associated signaling with the CS2K Core, such as QSIP queries.

The OM group NCASLINK in the MCS is associated with the OM group NCAS\_LINK in the CS2K Core. The OM group NCASLINK measures the performance of the NCAS link in the MCS. The OM group NCAS\_LINK measures the performance of the NCAS link in the CS2K Core.

The following table lists the key and info fields associated with OM group NCASLINK.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

The following functional groups are related to OM group NCASLINK:

- System Manager

### Registers

The following table lists the registers associated with OM group NCASLINK and what they measure. For a description of a register, click on the register name.

#### Registers for OM group NCASLINK

| Register name               | Measures               |
|-----------------------------|------------------------|
| <a href="#">linkUp</a>      | link up                |
| <a href="#">linkDown</a>    | link down              |
| <a href="#">msgSent</a>     | messages sent          |
| <a href="#">msgRcvd</a>     | messages received      |
| <a href="#">msgSendFail</a> | messages sent fail     |
| <a href="#">msgRcvFail</a>  | messages received fail |

**linkUp****Register type**

Peg

**Description**

The number of times the logical link is transitioned into a connected state due to administrative action, network repair, etc.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**linkDown****Register type**

Peg

**Description**

The number of times the logical link is transitioned into a disconnected state due to administrative action, network failure, etc.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**msgSent****Register type**

Peg

**Description**

The number of messages successfully sent over the link.

**Associated registers**[msgSendFail](#)**Extension registers**

None

**Associated logs**

None

**msgRcvd****Register type**

Peg

**Description**

The number of messages successfully received over the link.

**Associated registers**

[msgRcvFail](#)

**Extension registers**

None

**Associated logs**

None

**msgSendFail****Register type**

Peg

**Description**

The number of messages that failed to be sent.

**Associated registers**

None

**Extension registers**

[msgSent](#)

**Associated logs**

None

**msgRcvFail****Register type**

Peg

**Description**

The number of messages that failed to be received.

**Associated registers**

[msgRcvd](#)

**Extension registers**

None

**Associated logs**

None



---

## NCMCPUST

---

### Description

OM group Non-computing Module Node Central Processing Unit Status (NCMCPUST) provides information about the CPU occupancy of the following non-computing module (CM) node types:

- application processing unit (APU)
- CCS7 link interface unit (LIU7)
- high-speed link interface unit (HLIU)
- high-speed link router (HSLR)
- CCS7 Server (SVR7)
- Ethernet interface unit (EIU)
- Ethernet link interface unit (ELIU)
- frame relay interface unit (FRIU)
- X.25/X.75 link interface unit (XLIU)
- voice processing unit (VPU)

NCMCPUST uses registers that record the following CPU occupancies:

- call processing class
- scheduler class
- scheduler SYSTEM6 and SYSTEM7 class
- maintenance class
- non-guaranteed background class
- idler class
- input/output interrupt class

The CPU occupancy values accumulate at the non-CM node. The CPU occupancy values update at 1 min intervals during the transfer period. The values are collected from the CPSTATUS data. The accumulated CPU occupancy values transfer to the CM at the end of the transfer period. The CM copies this information into the operational measurements (OM) registers.

The following table lists the key and info fields associated with OM group NCMCPUST:

| Key field | Info field                                                                                                     |
|-----------|----------------------------------------------------------------------------------------------------------------|
| None      | LIU_type nnn; where LIU_type is EIU, ELIU, FRIU, HLIU, HSLR, LIU7, SVR7, or XLIU; and nnn is between 0 and 750 |

## Related functional groups

The functional group Ethernet Interface Unit is associated with OM group NCMCPUST.

## Registers

The following table lists the registers associated with OM group NCMCPUST and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group NCMCPUST

| Register name            | Measures                                     |
|--------------------------|----------------------------------------------|
| <a href="#">NCMBKG</a>   | Non-CM node background class occupancy       |
| <a href="#">NCMCPOCC</a> | Non-CM node call processing class occupancy  |
| <a href="#">NCMIDLE</a>  | Non-CM node idler class occupancy            |
| <a href="#">NCMIO</a>    | Non-CM node input/output interrupt occupancy |
| <a href="#">NCMMAINT</a> | Non-CM node maintenance class occupancy      |
| <a href="#">NCMSCHED</a> | Non-CM node scheduler class occupancy        |
| <a href="#">NCMSYS</a>   | Non-CM node system class occupancy           |

### NCMBKG

#### Register type

Peg

#### Description

NCMBKG records the CPU time its processes use and expresses the time as an integer. The processes are: the log system, audits, non-critical maintenance, OM accumulation and reporting.

The value NCMBKG records is the CPU background occupancy.

At the beginning of the transfer period, NCMBKG sets to zero. The CPU background occupancy values accumulate at the non-CM node and update at 1 min intervals during the transfer period. The values collect from the CPSTATUS data.

To obtain the average CPU background occupancy for 1 min, divide the holding register value by the transfer period (expressed in minutes).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMCPOCC****Register type**

Peg

**Description**

NCMCPOCC records the CPU time processing uses and expresses it as an integer.

At the beginning of the transfer period, NCMCPOCC sets to zero.

The CPU call processing occupancy values accumulate at the non-CM node. The values update at 1 min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU call processing occupancy for 1 min, divide the holding register value by the transfer period (expressed in minutes).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMIDLE****Register type**

Peg

**Description**

NCMIDLE records the CPU time the idler processes use and memory checks and expresses the time as an integer. The value that NCMIDLE records is the CPU idler occupancy. The CPU idler occupancy consists of the time that the processes use in the SYSTEM0 scheduler class.

At the beginning of the transfer period, NCMIDLE sets to zero. The CPU idler occupancy values accumulate at the non-CM node. The CPU idler occupancy values update at 1-min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU idler occupancy for 1 min, divide the holding register value by the transfer period (expressed in minutes).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMIO****Register type**

Peg

**Description**

NCMIO records the CPU time that service input/output interrupts use and expresses the time as an integer. The value that NCMIO records is the CPU input/output interrupt occupancy.

At the beginning of the transfer period, NCMIO sets to zero. The CPU input/output interrupt occupancy values accumulate at the non-CM node. The values update at 1-min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU input/output interrupt occupancy for 1 min, divide the holding register value by the transfer period.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMMAINT****Register type**

Peg

**Description**

NCMMAINT records the CPU time that critical system maintenance processes use and expresses the time as an integer. The value that register NCMMAINT records is the CPU maintenance occupancy. The CPU maintenance occupancy consists of the time processes use in the maintenance scheduler class.

At the beginning of the transfer period, register NCMMAINT sets to zero. The CPU maintenance occupancy values accumulate at the non-CM node. The values update at 1 min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU maintenance occupancy for 1 min, divide the holding register value by the transfer period.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMSCHED****Register type**

Peg

**Description**

NCMSCHED records the CPU time that the scheduler is in use and expresses the time as an integer.

At the beginning of the transfer period, NCMSCHED sets to zero. The CPU scheduler occupancy values accumulate at the non-CM node. The values update at 1 min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU scheduler occupancy for 1 min, divide the holding register value by the transfer period.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCMSYS****Register type**

Peg

**Description**

NCMSYS records the CPU time that system operations use and expresses the time as an integer. The value that NCMSYS records is the CPU system occupancy. The CPU system occupancy consists of the time processes in the SYSTEM6 and SYSTEM7 scheduler classes use.

At the beginning of the transfer period, NCMSYS sets to zero. The CPU system occupancy values accumulate at the non-CM node. The values update at 1-min intervals during the transfer period. The system collects the values from the CPSTATUS data.

To obtain the average CPU system occupancy for 1 min, divide the holding register value by the transfer period (expressed in minutes).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NDS0CARR

### Description

OM group 8-port 64-Kbps Non-multiplexed Digital Voice/data Carriers (NDS0CARR) counts the errors, faults and use for each NDS0 physical carrier.

The system generates logs for error and fault conditions. These conditions indicate a change in the alarm status of the extended multiprocessor system (XMS)-based peripheral module (XPM).

The following table lists the key and info fields associated with OM group NDS0CARR:

| Key field                          | Info field |
|------------------------------------|------------|
| MESSAGE_SWITCH_NUMBER<br>is 0 or 1 | NDS0OMINF  |

Enter the following fields in table CARRMTC: LOSRST, LOSOL, AISRST, AISOL, CLKLRST, CLKLOL, BPVLRST, BPVLOL, SLIPRST, and SLIPOL.

### Related functional groups

There are no functional groups related to OM group NDS0CARR.

### Registers

The following table lists the registers associated with OM group NDS0CARR and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NDS0CARR (Sheet 1 of 2)

| Register name            | Measures                               |
|--------------------------|----------------------------------------|
| <a href="#">CARCSBSY</a> | NDS0 physical carrier CBSY usage count |
| <a href="#">CARMBSY</a>  | NDS0 physical carrier ManB usage count |
| <a href="#">CARSBSY</a>  | NDS0 physical carrier SYSB usage count |
| <a href="#">ERRAIS</a>   | AIS error count                        |
| <a href="#">ERRBVRX</a>  | BPVRX error count                      |

**Registers for OM group NDS0CARR (Sheet 2 of 2)**

| Register name            | Measures           |
|--------------------------|--------------------|
| <a href="#">ERRBVTX</a>  | BPVTX error count  |
| <a href="#">ERRCLRXX</a> | CLKRX error count  |
| <a href="#">ERRCLTX</a>  | CLKTX error count  |
| <a href="#">ERRLOS</a>   | LOS error counts   |
| <a href="#">ERRSLRX</a>  | SLIPRX error count |
| <a href="#">ERRSLTX</a>  | SLIPTX error count |
| <a href="#">FLTAIS</a>   | AIS fault count    |
| <a href="#">FLTBVRX</a>  | BPVRX fault count  |
| <a href="#">FLTBVTX</a>  | BPVTX fault count  |
| <a href="#">FLTCLRXX</a> | CLKRX fault count  |
| <a href="#">FLTCLTX</a>  | CLKTX fault count  |
| <a href="#">FLTLOS</a>   | LOS fault count    |
| <a href="#">FLTSLRX</a>  | SLIPRX fault count |
| <a href="#">FLTSLTX</a>  | SLIPTX fault count |

**CARCSBSY****Register type**

Peg

**Description**

ARCSBSY samples the NDS0 physical carrier state every 100 seconds. Register CARCSBSY counts the time that the carrier is in the C-side busy (CBSY) state.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None



**CARMBSY****Register type**

Peg

**Description**

CARMBSY samples the NDS0 physical carrier state every 100 seconds. Register CARMBSY counts the time that the carrier is in the manually busy (ManB) state.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**CARSBSY****Register type**

Peg

**Description**

CARSBSY samples the NDS0 physical carrier state every 100 seconds. Register CARSBSY counts the time that the carrier is in the system busy (SYSB) state.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRAIS****Register type**

Peg

**Description**

ERRAIS counts the number of alarm indication signal (AIS) errors that occur. An AIS error occurs if a string of ones (1) is received on the receive data input.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRBVRX****Register type**

Peg

**Description**

ERRBVRX counts the number of bipolar violation loss on receive (BPVRX) errors that occur. The system detects a BPVRX error if a loss of 8 kHz violation occur in the clock used to receive data.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRBVTX****Register type**

Peg

**Description**

ERRBVTX counts the number of bipolar violation loss on receive (BPVRX) errors that occur. The system detects a BPVRX error if a loss of 8 kHz violation occur in the clock used to receive data.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRCLR****Register type**

Peg

**Description**

ERRCLR counts the number of receive clock loss (CLKRX) errors that occur. A CLKRX error occurs if the system detects loss of clock on the clock used to receive data.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRCLTX****Register type**

Peg

**Description**

ERRCLTX counts the number of transmit clock loss (CLKTX) errors that occur. A CLKTX error occurs if the system detects loss of clock on the clock used to transmit data.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRLOS****Register type**

Peg

**Description**

ERRLOS counts the number of loss-of-signal (LOS) errors that occur. A LOS error occurs if the system receives a stream of zeros (0) on the receive data input.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRSLRX****Register type**

Peg

**Description**

ERRSLRX counts the number of slip receive (SLIPRX) errors that occur. The system records an SLIPRX error. An SLIPRX error occurs when the rates at which the network transmits and receives data are different.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRSLTX****Register type**

Peg

**Description**

ERRSLTX counts the number of slip transmission (SLIPTX) errors that occur. When the system processes data at different rates, the system loses or repeats transmitted data and records a SLIPTX error. Processed data transmits or receives.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**FLTAIS****Register type**

Peg

**Description**

FLTAIS counts the number of alarm indication signal (AIS) faults that occur. A fault is an error that causes the carrier to become system busy (SYSB).

FLTAIS increases when the associated carrier becomes SYSB or when the AIS steady alarm raises. Register FLTAIS also increases when the AIS hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTBVRX****Register type**

Peg

**Description**

FLTBVRX counts the number of bipolar violation loss on receive (BPVRX) faults that occur.

FLTBVRX increases when the associated carrier becomes system busy (SYSB) or when the BPVRX steady alarm raises. Register FLTBVRX also increases when the BPVRX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTBVTX****Register type**

Peg

**Description**

FLTBVTX counts the number of bipolar violation loss on transmit (BPVTX) faults that occur.

FLTBVTX increases when the associated carrier becomes system busy (SYSB) or when the BPVTX steady alarm raises. Register FLTBVTX also increases when the BPVTX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTCLR****Register type**

Peg

**Description**

FLTCLR counts the number of receive clock loss (CLKRX) faults that occur.

FLTCLR increases when the associated carrier becomes system busy (SYSB) or when the CLKRX steady alarm raises. Register FLTCLR also increases when the CLKRX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTCLTX****Register type**

Peg

**Description**

FLTCLTX counts the number of transmit clock loss (CLKTX) faults that occur.

FLTCLTX increases when the associated carrier becomes system busy (SYSB) or when the CLKTX steady alarm raises. Register FLTCLTX also increases when the CLKTX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTLOS****Register type**

Peg

**Description**

FLTLOS counts the number of loss-of-signal (LOS) faults that occur.

FLTLOS increases when the associated carrier becomes system busy (SYSB) or when the LOS steady alarm raises. Register FLTLOS also increases when the LOS hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTSLRX****Register type**

Peg

**Description**

FLTSLRX counts the number of slip receive (SLIPRX) faults that occur.

FLTSLRX increases when the associated carrier becomes system busy (SYSB) or when the SLIPRX steady alarm raises. Register FLTSLRX also increases when the SLIPRX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187

**FLTSLTX****Register type**

Peg

**Description**

FLTSLTX counts the number of slip transmission (SLIPTX) faults that occur.

FLTSLTX increases when the associated carrier becomes system busy (SYSB) or when the SLIPTX steady alarm raises. Register FLTSLTX also increases when the SLIPTX hit-state alarm raises and the SETACTION field in table LTCPSINV is TRUE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM187



## NETMSG

### Description

OM group Network Message Service (NETMSG) monitors the use of network message services (NMS).

The OM group NETMSG contains four registers that count:

- NMS transaction capability application part (TCAP) requests that time out
- NMS TCAP requests that receive a negative acknowledgement
- invalid addresses from a message service
- NMS requests for an empty subscriber directory number

The following table lists the key and info fields associated with OM group NETMSG:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

CCS7 functional group is associated with OM group NETMSG.

### Registers

The following table lists the registers associated with OM group NETMSG and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NETMSG

| Register name            | Measures                           |
|--------------------------|------------------------------------|
| <a href="#">NMSDENL</a>  | Negative acknowledgement           |
| <a href="#">NMSINVAD</a> | Invalid address                    |
| <a href="#">NMSTIME</a>  | Time out                           |
| <a href="#">NMSVACT</a>  | Vacant subscriber directory number |

#### NMSDENL

**Register type**

Peg

**Description**

NMSDENL counts network NMS TCAP requests that receive negative acknowledgement.

A not having enough of 32-word FTRQ blocks, available at the server node, can affect register NMSDENL. The 32-word FTRQ blocks are in office parameter FTRQ32WAREAS, in table OFCENG.

Register NMSDENL increases at the host node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NMSINVAD****Register type**

Peg

**Description**

NMSINVAD counts addresses received from NMS that are not correct. An error can occur for two reasons. The message service agent can enter a directory number that is not correct. The NMS can generate a directory number that is not correct.

Register NMSINVAD increases at the host node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMS100

**NMSTIME****Register type**

Peg

**Description**

NMSTIME counts NMS TCAP requests that time out because the TCAP instruction disappears before it reaches the server node. This

register also counts NMS TCAP requests that time out. A TCAP request times out because the TCAP acknowledgement disappears before it reaches the host node.

Register NMSTIME increases at the host node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NMSVACT****Register type**

Peg

**Description**

NMSVACT counts NMS requests received for a empty subscriber directory number. The vacancy occurs for two reasons. The subscriber no longer exists at the server node, or the NMS generated an address that is possible but not correct.

Register NMSVACT increases at the server node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMS101

## NIUFBUS

### Description

OM group Network Interface Unit (NIU) Frame Transport Bus (F-bus) (NIUFBUS) monitors transmit and receive activity between the F-buses and the NIU.

The OM group NIUFBUS contains 30 two registers that count:

- number of packets an NIU transmits on each F-bus
- number of packets an NIU receives on each F-bus
- number of transmit errors an NIU makes on each F-bus
- number of receive errors an NIU makes on each F-bus
- number of octets an NIU transmits on each F-bus
- number of octets an NIU receives on each F-bus
- number of times an NIU turns on congestion on each F-bus
- number of high priority messages an NIU transmits on each F-bus
- number of messages that require placing in queue by an NIU on each F-bus

The following table lists the key and info fields associated with OM group NIUFBUS:

| Key field | Info field                                              |
|-----------|---------------------------------------------------------|
| None      | pm_type: NIU<br>pm_number: {integer}<br>pm_unit: {0..1} |

### Related functional groups

SuperNode DMS switch is associated with OM group NIUFBUS.

### Registers

The following table lists the registers associated with OM group NIUFBUS and what they measure. For a detailed description of a register,

click on the register name.

### Registers for OM group NIUFBUS

| Register name            | Measures                               |
|--------------------------|----------------------------------------|
| <a href="#">NF0RXERR</a> | Frame transport bus 0 receive errors   |
| <a href="#">NF0RXOCT</a> | Frame transport bus 0 receive octets   |
| <a href="#">NF0RXPKT</a> | Frame transport bus 0 receive packets  |
| <a href="#">NF0TXCON</a> | F-bus 0 transmit congestion            |
| <a href="#">NF0TXENQ</a> | F-bus 0 transmit placing in queue      |
| <a href="#">NF0TXERR</a> | Frame transport bus 0 transmit errors  |
| <a href="#">NF0TXOCT</a> | Frame transport bus 0 transmit octets  |
| <a href="#">NF0TXPKT</a> | Frame transport bus 0 transmit packets |
| <a href="#">NF0TXPRI</a> | F-bus 0 transmit priority              |
| <a href="#">NF1RXERR</a> | Frame transport bus 1 receive errors   |
| <a href="#">NF1RXOCT</a> | Frame transport bus 1 receive octets   |
| <a href="#">NF1RXPKT</a> | Frame transport bus 1 receive packets  |
| <a href="#">NF1TXCON</a> | F-bus 1 transmit congestion            |
| <a href="#">NF1TXENQ</a> | F-bus 1 transmit placing in queue      |
| <a href="#">NF1TXERR</a> | Frame transport bus 1 transmit errors  |
| <a href="#">NF1TXOCT</a> | Frame transport bus 1 transmit octets  |
| <a href="#">NF1TXPKT</a> | Frame transport bus 1 transmit packets |
| <a href="#">NF1TXPRI</a> | F-bus 1 transmit priority              |

#### NF0RXERR

**Register type**

Peg

**Description**

NF0RXERR increases in an audit period by the number of packets that an NIU did not receive on Fbus. The NIU did not receive the packets because of an error.

**Associated registers**

None

**Extension registers**

NF0RXER2

**Associated logs**

None

**NF0RXOCT****Register type**

Peg

**Description**

NF0RXOCT increases by the number of octets (bytes) an NIU receives on Fbus 0.

**Associated registers**

None

**Extension registers**

NF0RXOC2

**Associated logs**

None

**NF0RXPKT****Register type**

Peg

**Description**

NF0RXPKT increases in an audit period by the number of packets an NIU receives from Fbus 0.

**Associated registers**

None

**Extension registers**

NF0RXP2

**Associated logs**

None

**NF0TXCON****Register type**

Peg

**Description**

NF0TXCON counts the number of times the NIU turns on congestion for F-bus 0.

**Associated registers**

[NF1TXCON](#)

**Extension registers**

None

**Associated logs**

None

**NF0TXENQ****Register type**

Peg

**Description**

NF0TXENQ counts the number of messages on F-bus 0 that require placing in queue.

**Associated registers**

[NF1TXENQ](#)

**Extension registers**

NF0TXEN2

**Associated logs**

None

**NF0TXERR****Register type**

Peg

**Description**

NF0TXERR increases in an audit period by the number of packets an NIU could not send out on Fbus. The NIU could not send the packets because of an error.

**Associated registers**

None

**Extension registers**

NF0TXER2

**Associated logs**

None

**NF0TXOCT****Register type**

Peg

**Description**

NF0TXOCT increases the number of octets (bytes) an NIU transmits on Fbus 0.

**Associated registers**

None

**Extension registers**

NF0TXOC2

**Associated logs**

None

**NF0TXPKT****Register type**

Peg

**Description**

NF0TXPKT increases in an audit period by the number of packets an NIU transmits on Fbus 0.

**Associated registers**

None

**Extension registers**

NF0TXPK2

**Associated logs**

None

**NF0TXPRI****Register type**

Peg



**Description**

NF0TXPRI counts the number of high priority messages that are transmitted on F-bus 0.

**Associated registers**

[NF1TXPRI](#)

**Extension registers**

None

**Associated logs**

None

**NF1RXERR****Register type**

Peg

**Description**

NF1RXERR increases in an audit period by the number of packets an NIU did not receive on Fbus 1. The NIU did not receive the packets because of an error.

**Associated registers**

None

**Extension registers**

NF1RXER2

**Associated logs**

None

**NF1RXOCT****Register type**

Peg

**Description**

NF1RXOCT increases by the number of octets (bytes) an NIU receives on Fbus 1.

**Associated registers**

None

**Extension registers**

NF1RXOC2

**Associated logs**

None

**NF1RXPKT****Register type**

Peg

**Description**

NF1RXPKT increases in an audit period by the number of packets an NIU receives from Fbus 1.

**Associated registers**

None

**Extension registers**

NF1RXPK2

**Associated logs**

None

**NF1TXCON****Register type**

Peg

**Description**

NF1TXCON counts the number of times an NIU turns on congestion for F-bus 1.

**Associated registers**[NF0TXCON](#)**Extension registers**

None

**Associated logs**

None

**NF1TXENQ****Register type**

Peg

**Description**

NF1TXENQ counts the number of messages on F-bus 1 that require placing in queue.

**Associated registers**[NF0TXENQ](#)**Extension registers**

NF1TXEN2

**Associated logs**

None

**NF1TXERR****Register type**

Peg

**Description**

NF1TXERR increases in an audit period by the number of packets an NIU could not send out on Fbus 1. The NIU could not send the packets because of an error.

**Associated registers**

None

**Extension registers**

NF1TXER2

**Associated logs**

None

**NF1TXOCT****Register type**

Peg

**Description**

NF1TXOCT increases the number of octets (bytes) an NIU transmits on Fbus 1.

**Associated registers**

None

**Extension registers**

NF1TXOC2

**Associated logs**

None

**NF1TXPKT****Register type**

Peg

**Description**

NF1TXPKT increases in an audit period by the number of packets an NIU transmits on Fbus 1.

**Associated registers**

None

**Extension registers**

NF1TXPK2

**Associated logs**

None

**NF1TXPRI****Register type**

Peg

**Description**

NF1TXPRI counts the number of high priority messages that are transmitted on F-bus 1.

**Associated registers**[NF0TXPRI](#)**Extension registers**

None

**Associated logs**

None

## NIUMEMUT

### Description

OM group Network Interface Unit Memory Use (NIUMEMUT) displays data and program store information for a network interface unit (NIU).

The NIUMEMUT contains four registers that:

- hold the total data store memory
- hold the free data store memory
- hold the total program store memory
- hold the free program store memory

The following table lists the key and info fields associated with OM group NIUMEMUT:

| Key field | Info field                                              |
|-----------|---------------------------------------------------------|
| None      | PM_TYPE: NIU<br>PM NUMBER: {integer}<br>PM_UNIT: {0..1} |

### Related functional groups

There are no functional groups associated with OM group NIUMEMUT.

### Registers

The following table lists the registers associated with OM group NIUMEMUT and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NIUMEMUT

| Register name            | Measures                                               |
|--------------------------|--------------------------------------------------------|
| <a href="#">NIUSTOT</a>  | Network interface unit total data store (DS) memory    |
| <a href="#">NIUDSAVL</a> | Network interface unit free DS memory                  |
| <a href="#">NIUPSTOT</a> | Network interface unit total program store (PS) memory |
| <a href="#">NIUPSAVL</a> | Network interface unit free PS memory                  |

**NIUSTOT****Register type**

Peg

**Description**

NIUSTOT holds the total data store memory information in Kbytes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NIUDSAVL****Register type**

Peg

**Description**

NIUDSAVL holds available DS memory information in Kbytes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NIUPSTOT****Register type**

Peg

**Description**

NIUPSTOT holds the total PS memory information in Kbytes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NIUPSAVL****Register type**

Peg

**Description**

NIUPSAVL holds the available PS memory information in Kbytes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NMC

### Description

OM group Network Module Controller Maintenance Summary (NMC) counts errors and failures to recover from errors in the following:

- in-service message links between network modules and peripheral modules
- speech connections
- in-service network module controllers

The OM group NMC also records if out-of-service network modules, network module ports, and junctors are system busy or manual busy.

All the measurements in NMC refer to individual components, not paired duplicates. The failures recorded in NMC do not always indicate lost calls.

The OM group NMC contains six peg registers and six usage registers. Scan rate for the usage registers is slow: 100 seconds.

The OM group NMC used to analyze network module controller maintenance.

All DMS offices have an OM group NMC.

The following table lists the key and info fields associated with OM group NMC:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group NMC.

### Registers

The following table lists the registers associated with OM group NMC and what they measure. For a detailed description of a register, click



on the register name.

### Registers for OM group NMC

| Register name             | Measures                                  |
|---------------------------|-------------------------------------------|
| <a href="#">NMCERR</a>    | Network module controller errors          |
| <a href="#">NMCFLT</a>    | Network module controller failure         |
| <a href="#">NMJRMBU</a>   | Network module junctors manual busy usage |
| <a href="#">NMJRMBU</a>   | Network module junctors system busy usage |
| <a href="#">NMMBU</a>     | Network module manual busy usage          |
| <a href="#">NMMSGERR</a>  | Network module message link errors        |
| <a href="#">NMMSGFL</a>   | Network module message link failures      |
| <a href="#">NMPTMBU</a>   | Network module ports manual busy usage    |
| <a href="#">NMPTSBU</a>   | Network module ports system busy usage    |
| <a href="#">NMSBU</a>     | Network modules system busy usage         |
| <a href="#">NMSPCHERR</a> | Network module speech connection errors   |
| <a href="#">NMSPCHFL</a>  | Network module speech connection failure  |

#### NMCERR

##### Register type

Peg

##### Description

NMCERR counts errors that are in in-service network module controllers.

##### Associated registers

[NMCFLT](#)

##### Extension registers

None

##### Associated logs

NETM128

**NMCFLT****Register type**

Peg

**Description**

NMCFLT increases when a network module controller cannot recover from an error. The controller remains system busy, pending manual maintenance or a successful system-initiated recovery.

**Associated registers**[NMCERR](#)**Extension registers**

None

**Associated logs**

NETM112, NETM128, NETM116, METM120, NETM122

**NMJRMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMJRMBU records if network module junctors are manual busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM123, NETM140

**NMJRSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMJRSBU records if network module junctors are system busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM122

**NMMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMMBU records if out-of-service network modules are manual busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM105, NETM138

**NMMSGGER****Register type**

Peg

**Description**

NMMSGGER counts errors in in-service message links between network modules and peripheral modules.

**Associated registers**[NMMSGFL](#)**Extension registers**

None

**Associated logs**

NET102, NTEM129

**NMMSGFL****Register type**

Peg

**Description**

NMMSGFL increases when a link between a network module and a peripheral module cannot recover from an error. The link remains system busy, pending manual maintenance or a successful system-initiated recovery attempt.

**Associated registers**[NMMSGER](#)**Extension registers**

None

**Associated logs**

NETM120, NETM126, NETM129

**NMPTMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMPTMBU records if network module ports are manual busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM117, NETM139

**NMPTSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMPTSBU records if network module ports are system busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM116, NETM129

**NMSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NMMSGFL records if out-of service network modules are system busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NETM103, NETM104

**NMSPCHER****Register type**

Peg

**Description**

NMSPCHER counts errors in speech connections located in the network.

**Associated registers**

[NMSPCHFL](#)

**Extension registers**

None

**Associated logs**

NET102

**NMSPCHFL****Register type**

Peg

**Description**

NMSPCHFL counts faults that in the network-resident connection memory, or in a speech path segment that is internal to the network. An accuracy failure that register NMSPCHFL recorded earlier trippers tests that detects the fault. The path segment affected is not available for call processing.

**Associated registers**[NMSPCHER](#)**Extension registers**

None

**Associated logs**

NET102, NETM120, NETM126, NETM129, NET131

## NMSH323

### Description

OM group NMSH323 measures the PRI messages sent by the CS2K Core to the H.323 GWC and received by the core from the H.323 GWC when passing through PRA Selector of the table MSGRTE.

This OM group accurately tracks the message traffic between CS2K Core and H.323 GWC.

The following table lists the key and info fields associated with OM group NMSH323.

| Key field | Info field                                   |
|-----------|----------------------------------------------|
| NMSH323   | NMSFACST<br>NMSREJST<br>NMSFACRV<br>NMSREJRV |

### Related functional groups

There are no functional groups associated with OM group NMSNCAS.

### Registers

The following table lists the registers associated with OM group NMSH323 and what they measure. For a description of a register, click on the register name.

#### Registers for OM group NMSH323

| Register name | Measures                                                          |
|---------------|-------------------------------------------------------------------|
| NMSFACST      | Records the number of NMS FAC messages sent to H323 GWC           |
| NMSREJST      | Record the number of NMS FACREJ messages sent to H323 GWC         |
| NMSFACRV      | Records the number of NMS FAC messages received from H323 GWC     |
| NMSREJRV      | Records the number of NMS FAC REJ messages received from H323 GWC |

**NMSFACST****Register type**

Peg

**Description**

NMSFACST records the number of NMS FAC message that have been sent to the H.323 GWC. This register applies only to DMS.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMSH111

**NMSREJST****Register type**

Peg

**Description**

NMSREJST records the number of NMS FAC REJ messages sent to the H.323 WGWC. This register applies only to DMS.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMSH112

**NMSFACRV****Register type**

Peg

**Description**

NMSFACRV records the number of NMS FAC messages received from the H.323 GWC by the CORE. This register applies to DMS only.

**Associated registers**

None

**Extension registers**

None



**Associated logs**

NMSH113

**NMSREJRV****Register type**

Peg

**Description**

NMSREJRV records the number of NMS FAC REJ messages received from the H.323 GWC. This register applies only to DMS.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMSH114

## NMSNCAS

### Description

OM group NMSNCAS keeps a record of the NMS messages sent and received by the CS2K Core over a NCAS link.

This OM group accurately tracks the messages sent and received on the NCAS link between CS2K Core and Session Server. The OM pegging provides information on the message traffic between CS2K Core and Session Server.

The following table lists the key and info fields associated with OM group NMSNCAS.

| Key field | Info field |
|-----------|------------|
|           |            |

### Related functional groups

There are no functional groups associated with OM group NMSNCAS.

### Registers

The following table lists the registers associated with OM group NMSNCAS and what they measure. For a description of a register, click on the register name.

#### Registers for OM group NMSNCAS

| Register name            | Measures                                          |
|--------------------------|---------------------------------------------------|
| <a href="#">SCTPNMSS</a> | NMS TCAP messages sent successfully over SCTP     |
| <a href="#">SCTPNMSR</a> | NMS TCAP messages received successfully over SCTP |
| <a href="#">SCTPREJS</a> | NMS REJ messages sent successfully over SCTP      |
| <a href="#">SCTPREJR</a> | NMS REJ messages received successfully over SCTP  |

#### SCTPNMSS

##### Register type

Peg

**Description**

SCTPMSS counts how many NMS TCAP messages are sent to Sctp in a half hour time period. This will provide information of performance needs for the NCAS link.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMSS115

**SCTPNMSR****Register type**

Peg

**Description**

SCTPNMSR counts how many NMS TCAP messages are received from Sctp in a half hour time period. This will provide information of performance needs for the NCAS link.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NMSS116

**SCTPREJS****Register type**

Peg

**Description**

SCTPREJS counts how many NMS REJECT messages are sent to Sctp in a half hour time period.

**Associated registers**

None

**Extension registers**

None

**Associated logs**  
NMSS117

**SCTPREJR**

**Register type**  
Peg

**Description**

SCTPREJR counts how many NMS REJECT messages are received from SCTP in a half hour time period.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
NMSS118

## NMTCLINK

### Description

OM group Node Maintenance - Link (NMTCLINK) measures the performance of transport media to a node that affects the maintenance reliability of this node. The data indicates the number of system troubles and out-of-service occurrences.

The following table lists the key and info fields associated with OM group NMTCLINK.

| Key field | Info field         |
|-----------|--------------------|
| None      | INM_OM_LINK_INFO_T |

### Related functional groups

There are no functional groups associated with OM group NMTCLINK.

### Registers

The following table lists the registers associated with OM group NMTCLINK and what they measure. For a description of a register, click on the register name.

**Note:** The OM group NMTCLINK provides one tuple for each node.

#### Registers for OM group NMTCLINK (Sheet 1 of 2)

| Register name            | Measures                                                |
|--------------------------|---------------------------------------------------------|
| <a href="#">NDMCHERR</a> | Node maintenance message channel errors                 |
| <a href="#">NDMCHFLT</a> | Node maintenance message channel faults                 |
| <a href="#">NDMCHMBP</a> | Node maintenance message channel manual busy (ManB) peg |
| <a href="#">NDMCHSBP</a> | Node maintenance message channel SYSB peg               |
| <a href="#">NDPLKERR</a> | Node maintenance physical link errors                   |
| <a href="#">NDPLKFLT</a> | Node maintenance physical link faults                   |

**Registers for OM group NMTCLINK (Sheet 2 of 2)**

| Register name            | Measures                                              |
|--------------------------|-------------------------------------------------------|
| <a href="#">NDPLKMBP</a> | Node maintenance physical link manual busy (ManB) peg |
| <a href="#">NDPLKSBP</a> | Node maintenance physical link system-busy (SysB) peg |

**NDMCHERR****Register type**

Peg

**Description**

NDMCHERR counts the number of errors in all important message channels to a node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDMCHFLT****Register type**

Peg

**Description**

NDMCHFLT counts the number of errors that persist after execution of diagnostics on important message channels. The register increases if the first diagnostic attempt does not clear the error. More tests of the error condition do not increase the register.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDMCHMBP****Register type**

Peg

**Description**

NDMCHMBP counts the times message channels become ManB.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDMCHSBP****Register type**

Peg

**Description**

NDMCHSBP counts the times message channels become SYSB.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDPLKERR****Register type**

Peg

**Description**

NDPLKERR counts the errors detected in all important physical channels to a node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDPLKFLT****Register type**

Peg

**Description**

NDPLKFLT counts the errors that persist after execution of diagnostics on important physical channels. The register increases if the first diagnostic attempt does not clear the error. More tests of the error do not increase the register.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDPLKMBP****Register type**

Peg

**Description**

NDPLKMBP counts the times physical channels become ManB.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDPLKSBP****Register type**

Peg

**Description**

NDPLKSBP counts the times physical channels become SysB.



**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NMTCNODE

### Description

OM group Node Maintenance - Node (NMTCNODE) measures the maintenance reliability performance of a node. The data shows the number of system troubles and out-of-service occurrences.

NMTCNODE is valid only for sync-matched nodes, which operate in synchronous mode. An example of this design is file processors on an SCP11. While the simplex mode is not in sync, consider the simplex mode of operation is in-service trouble (ISTb). Registers in NMTCNODE measure the amount of time the node is in this state.

The following table lists the key and info fields associated with OM group NMTCNODE.

| Key field | Info field         |
|-----------|--------------------|
| None      | INM_OM_NODE_INFO_T |

### Related functional groups

There are no functional groups associated with OM group NMTCNODE.

### Registers

The following table lists the registers associated with OM group NMTCNODE and what they measure. For a description of a register, click on the register name.

#### Registers for OM group NMTCNODE (Sheet 1 of 2)

| Register name          | Measures                                       |
|------------------------|------------------------------------------------|
| <a href="#">NDNERR</a> | Node maintenance node errors                   |
| <a href="#">NDNFLT</a> | Node maintenance node faults                   |
| <a href="#">NDNLRP</a> | Node maintenance node loss of redundancy peg   |
| <a href="#">NDNLRU</a> | Node maintenance node loss of redundancy usage |
| <a href="#">NDNMBP</a> | Node maintenance node manual busy (ManB) peg   |

**Registers for OM group NMTCNODE (Sheet 2 of 2)**

| Register name                    | Measures                                            |
|----------------------------------|-----------------------------------------------------|
| <a href="#"><u>NDNMBU</u></a>    | Node maintenance node ManB usage                    |
| <a href="#"><u>NDNMCRST</u></a>  | Node maintenance node manual cold restarts          |
| <a href="#"><u>NDNMCXFRN</u></a> | Node maintenance node manual-controlled transfers   |
| <a href="#"><u>NDNMRRST</u></a>  | Node maintenance node manual reload restarts        |
| <a href="#"><u>NDNMWRST</u></a>  | Node maintenance node manual warm restarts          |
| <a href="#"><u>NDNNAP</u></a>    | Node maintenance node not-available peg             |
| <a href="#"><u>NDNNAU</u></a>    | Node maintenance node not-available usage           |
| <a href="#"><u>NDNSBP</u></a>    | Node maintenance node system-busy (SysB) peg        |
| <a href="#"><u>NDNSBU</u></a>    | Node maintenance node SysB usage                    |
| <a href="#"><u>NDNSCRST</u></a>  | Node maintenance node system-controlled restarts    |
| <a href="#"><u>NDNSCXFR</u></a>  | Node maintenance node system-controlled transfers   |
| <a href="#"><u>NDNSRRST</u></a>  | Node maintenance node system reload restarts        |
| <a href="#"><u>NDNSUXFR</u></a>  | Node maintenance node system uncontrolled transfers |
| <a href="#"><u>NDNSWERR</u></a>  | Node maintenance node software errors               |
| <a href="#"><u>NDNSWRST</u></a>  | Node maintenance node system warm restarts          |
| <a href="#"><u>NDNTRAP</u></a>   | Node maintenance node trap errors                   |

**NDNERR****Register type**

Peg

**Description**

NDNERR counts the number of errors in an in-service or out-of-service node. NDNERR counts errors if more action either is or is not taken to correct the errors. The events can range from one-time hits to total

failures. NDNERR increases when a node application reports an error that results from manual or system maintenance.

**Associated registers**

[NDNSWERR](#), [NDNTRAP](#)

**Extension registers**

None

**Associated logs**

None

**NDNFLT****Register type**

Peg

**Description**

NDNFLT counts the number of errors that persist after diagnostics are executed. The fault register increases only when the first diagnostic attempt does not clear the error. Additional tests of the error condition do not increase the fault register.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNLRP****Register type**

Peg

**Description**

NDNLRP counts the number of peripherals that have one unit in service while another unit goes out of service. A count of zero occurs in NDNLRP if all units of the node are in service.

**Associated registers**

[NDNLRU](#)

**Extension registers**

None

**Associated logs**

None

**NDNLRU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDNLRU measures the length of time that the node has one unit in service while another unit is out of service. The count is based on a sample of the node that the system takes every 100 seconds.

**Associated registers**[NDNLRP](#)**Extension registers**

None

**Associated logs**

None

**NDNMBP****Register type**

Peg

**Description**

NDNMBP counts the number of times that a node goes into the manual busy (ManB) state.

**Associated registers**[NDNMBU](#)**Extension registers**

None

**Associated logs**

None

**NDNMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDNMBU counts the length of time that the node is in the manual busy (ManB) state. The count is based on a sample of the node that the system takes every 100 seconds.

**Associated registers**

[NDNMBP](#)

**Extension registers**

None

**Associated logs**

None

**NDNMCRST****Register type**

Peg

**Description**

NDNMCRST counts the number of cold restarts that occur on a node as the result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNMCXFRN****Register type**

Peg

**Description**

NDNMCXFR counts the number of times that a node switches activity as a result of manual action (controlled switch of activity). The count is correct for sync-matched node designs only. Other node designs have a count that is always zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNMRRST****Register type**

Peg

**Description**

NDNMRRST counts the number of reload restarts that occur on a node as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNMWRST****Register type**

Peg

**Description**

NDNMWRST counts the number of warm restarts that occur on a node as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNNAP****Register type**

Peg

**Description**

NDNNAP counts the number of times a node is isolated from the DMS.

**Associated registers**[NDNNAU](#)**Extension registers**

None

**Associated logs**

None

**NDNNAU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDNNAU measures the length of time the node is isolated from the DMS. The count is based on a sample of the node that the system takes every 100 seconds.

**Associated registers**[NDNNAP](#)**Extension registers**

None

**Associated logs**

None

**NDNSBP****Register type**

Peg

**Description**

NDNSBP counts the number of times a node goes into the SysB state.

**Associated registers**[NDNSBU](#)**Extension registers**

None

**Associated logs**

None



**NDNSBU****Register type**

Usage

**Scan rate**

Not available

**Description**

NDNSBU measures the length of time that a node is SysB.

**Associated registers**[NDNSBP](#)**Extension registers**

None

**Associated logs**

None

**NDNSCRST****Register type**

Peg

**Description**

NDNSCRST counts the number of times a cold restart occurs on a node as the result of a system operation.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNSCXFR****Register type**

Peg

**Description**NDNSCXFR counts the number of times a node switches activity as the result of a controlled system maintenance operation. *Controlled* means that the node maintenance system is able to prepare for the switch of activity before it occurs. The count applies to sync-matched node designs only. The count for other node designs is always zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNSRRST****Register type**

Peg

**Description**

NDNSRRST counts the number of reload restarts that occur on a node as the result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNSUXFR****Register type**

Peg

**Description**

NDNSUXFR counts the number of times a node switches activity as the result of uncontrolled system maintenance action. *Uncontrolled* means that the node maintenance cannot prepare for the switch of activity before it happens. The count applies to sync-matched node designs only. The count is always zero for other node designs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNSWERR****Register type**

Peg

**Description**

NDNSWERR counts the number of times a software error occurs on a node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNSWRST****Register type**

Peg

**Description**

NDNSWRST counts the number of warm restarts that occur on a node as the result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDNTRAP****Register type**

Peg

**Description**

NDNTRAP counts the number of trap errors that occur on a node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**  
None

## NMTCTYPE

### Description

OM group Node Maintenance - Node Type (NMTCTYPE) measures the performance of the nodes in each node type. The data shows the number of system problems and out-of-service occurrences. The system uses register values in OM group NMTCNODE to generate the values in NMTCTYPE.

NMTCTYPE is correct only for sync-matched nodes, which operate in synchronous mode. An example of sync-matched nodes is file processors on an SCPII. While the simplex mode is not in sync:

- the simplex mode of operation is considered to be in-service trouble (ISTb)
- the registers in this group measure the amount of time the node spends in this state

The following table lists the key and info fields associated with OM group NMTCTYPE.

| Key field        | Info field         |
|------------------|--------------------|
| INM_NODE_CLASS_T | INM_OM_TYPE_INFO_T |

### Related functional groups

There are no functional groups associated with OM group NMTCTYPE.

### Registers

The following table lists the registers associated with OM group NMTCTYPE and what they measure. For a description of a register, click on the register name.

**Note:** OM group NMTCTYPE provides one tuple for each node type (maximum 1023).

#### Registers for OM group NMTCTYPE (Sheet 1 of 2)

| Register name          | Measures                                     |
|------------------------|----------------------------------------------|
| <a href="#">NDTERR</a> | Node maintenance type errors                 |
| <a href="#">NDTFLT</a> | Node maintenance type faults                 |
| <a href="#">NDTLRP</a> | Node maintenance type loss of redundancy peg |

**Registers for OM group NMTCTYPE (Sheet 2 of 2)**

| Register name                   | Measures                                            |
|---------------------------------|-----------------------------------------------------|
| <a href="#"><u>NDTLRU</u></a>   | Node maintenance type loss of redundancy usage      |
| <a href="#"><u>NDTMBP</u></a>   | Node maintenance type manual-busy (ManB) peg        |
| <a href="#"><u>NDTMBU</u></a>   | Node maintenance type ManB usage                    |
| <a href="#"><u>NDTMCRST</u></a> | Node maintenance type manual cold restarts          |
| <a href="#"><u>NDTMCXFR</u></a> | Node maintenance type manual-controlled transfers   |
| <a href="#"><u>NDTMRRST</u></a> | Node maintenance type manual reload restarts        |
| <a href="#"><u>NDTMWRST</u></a> | Node maintenance type manual warm restarts          |
| <a href="#"><u>NDTNAP</u></a>   | Node maintenance type not-available peg             |
| <a href="#"><u>NDTNAU</u></a>   | Node maintenance type not-available usage           |
| <a href="#"><u>NDTSBP</u></a>   | Node maintenance type system-busy (SysB) peg        |
| <a href="#"><u>NDTSBU</u></a>   | Node maintenance type SysB usage                    |
| <a href="#"><u>NDTSCRST</u></a> | Node maintenance type system cold restart           |
| <a href="#"><u>NDTSCXFR</u></a> | Node maintenance type system-controlled transfers   |
| <a href="#"><u>NDTSRRST</u></a> | Node maintenance type system reload restarts        |
| <a href="#"><u>NDTSUXFR</u></a> | Node maintenance type system uncontrolled transfers |
| <a href="#"><u>NDTSWERR</u></a> | Node maintenance type software errors               |
| <a href="#"><u>NDTSWRST</u></a> | Node maintenance type system warm restarts          |
| <a href="#"><u>NDTTRAP</u></a>  | Node maintenance type trap errors                   |

**NDTERR****Register type**

Peg

**Description**

NDTERR counts the number of errors detected on in-service or out-of-service nodes for each node type. NDTERR counts errors whether or not the errors receive additional action. The errors counted range from one-time hits to total failures. The register increases when an application on a node reports an error that is a result of a manual or system maintenance action.

**Associated registers**

[NDTSWERR](#), [NDTTRAP](#)

**Extension registers**

None

**Associated logs**

None

**NDTFLT****Register type**

Peg

**Description**

NDTFLT counts the number of errors remaining after diagnostics. The register increases if the first diagnostic attempt does not clear the error. Additional tests on the error condition do not cause the register to increase.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTLRP****Register type**

Peg

**Description**

NDTLRP counts the number of peripherals that have one unit in service while another unit goes out of service. The register counts zero if all units of the node type are in service.

**Associated registers**[NDTLRU](#)**Extension registers**

None

**Associated logs**

None

**NDTLRU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDTLRU measures the length of time that each node type has one unit in service while another unit is out of service. The count is based on node samples the system takes every 100 seconds.

**Associated registers**[NDTLRP](#)**Extension registers**

None

**Associated logs**

None

**NDTMBP****Register type**

Peg

**Description**

NDTMBP counts the number of times that each node type becomes ManB.

**Associated registers**[NDTMBU](#)**Extension registers**

None

**Associated logs**

None



**NDTMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDTMBU measures the length of time that each node type becomes ManB. The count is based on node samples taken every 100 seconds.

**Associated registers**

[NDTMBP](#)

**Extension registers**

None

**Associated logs**

None

**NDTMCRST****Register type**

Peg

**Description**

NDTMCRST counts the number of cold restarts that occur on each node type as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTMCXFR****Register type**

Peg

**Description**

NDTMCXFR counts the number of times that each node type switches activity as a result of manual action. The register classifies a manual transfer as a controlled switch. The count applies to sync-matched nodes only. For other node designs, the count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTMRRST**

NDTMRRST counts the number of reload-restarts that occur on each node type as a result of manual action.

**Register type**

Peg

**Description**

NDTMRRST counts the number of reload-restarts that occur on each type of node because of manual operations.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTMWRST****Register type**

Peg

**Description**

NDTMWRST counts the number of warm restarts that occur on each node type as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTNAP****Register type**

Peg

**Description**

NDTNAP counts the number of times the system isolates each node type from the DMS.

**Associated registers**[NDTNAU](#)**Extension registers**

None

**Associated logs**

None

**NDTNAU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDTNAU measures the length of time that the system isolates each node type from the DMS. The count is based on node samples the system takes every 100 seconds.

**Associated registers**[NDTNAP](#)**Extension registers**

None

**Associated logs**

None

**NDTSBP****Register type**

Peg

**Description**

NDTSBP counts the number of times each node type becomes SysB.

**Associated registers**[NDTSBU](#)**Extension registers**

None

**Associated logs**

None

**NDTSBU****Register type**

Usage

**Scan rate**

Not available

**Description**

NDTSBU measures the length of time each node type is SysB.

**Associated registers**[NDTSBP](#)**Extension registers**

None

**Associated logs**

None

**NDTSCRST****Register type**

Peg

**Description**

NDTSCRST counts the number of times a cold restart occurs on each node type as a result of a system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTSCXFR****Register type**

Peg

**Description**

NDTSCXFR counts the number of times each node type switches activity as a result of a controlled system maintenance action.

*Controlled* means that node maintenance can prepare for the switch of activity before it occurs. The count applies to sync-matched nodes only. For other node designs, the count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTSRRST****Register type**

Peg

**Description**

NDTSRRST counts the number of reload restarts that occur on each node type as a result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTSUXFR****Register type**

Peg

**Description**

NDTSUXFR counts the number of times each node type switches activity as a result of an uncontrolled system maintenance action.

*Uncontrolled* means that node maintenance cannot prepare for the

switch of activity before it occurs. The count applies to sync-matched nodes. For other nodes, the count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTSWERR****Register type**

Peg

**Description**

NDTSWERR counts the number of times a software error occurs on each node type.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTSWRST****Register type**

Peg

**Description**

NDTSWRST counts the number of warm restarts that occur on each node type as a result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDTTRAP****Register type**

Peg

**Description**

NDTTRAP counts the number of traps that occur on each node type.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NMTCUNIT

### Description

OM group Node Maintenance - Unit (NMTCUNIT) measures the maintenance reliability performance of one unit of a node. The data shows the number of system problems and out-of-service occurrences. The group does not apply to sync-matched nodes, which operate in synchronous mode. File processors on an SCP II are examples of sync-matched nodes.

The following table lists the key and info fields associated with OM group NMTCUNIT.

| Key field | Info field         |
|-----------|--------------------|
| None      | INM_OM_UNIT_INFO_T |

### Related functional groups

There are no functional groups associated with OM group NMTCUNIT.

### Registers

The following table lists the registers associated with OM group NMTCUNIT and what they measure. For a description of a register, click on the register name.

**Note:** OM group NMTCUNIT provides two tuples for each node.

#### Registers for OM group NMTCUNIT (Sheet 1 of 2)

| Register name            | Measures                                     |
|--------------------------|----------------------------------------------|
| <a href="#">NDUERR</a>   | Node maintenance unit errors                 |
| <a href="#">NDUFLT</a>   | Node maintenance unit faults                 |
| <a href="#">NDUMBP</a>   | Node maintenance unit manual-busy (ManB) peg |
| <a href="#">NDUMBU</a>   | Node maintenance unit ManB usage             |
| <a href="#">NDUMCRST</a> | Node maintenance unit manual cold restarts   |
| <a href="#">NDUMRRST</a> | Node maintenance unit manual reload restarts |
| <a href="#">NDUMWRST</a> | Node maintenance unit manual warm restarts   |



**Registers for OM group NMTCUNIT (Sheet 2 of 2)**

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">NDUNAP</a>   | Node maintenance unit not-available peg          |
| <a href="#">NDUNAU</a>   | Node maintenance unit not-available usage        |
| <a href="#">NDUSBP</a>   | Node maintenance unit system-busy (SysB) peg     |
| <a href="#">NDUSBU</a>   | Node maintenance unit SysB usage                 |
| <a href="#">NDUSCRST</a> | Node maintenance unit system-controlled restarts |
| <a href="#">NDUSRRST</a> | Node maintenance unit system reload restarts     |
| <a href="#">NDUSWERR</a> | Node maintenance unit software errors            |
| <a href="#">NDUSWRST</a> | Node maintenance unit system warm restarts       |
| <a href="#">NDUTRAP</a>  | Node maintenance unit trap errors                |

**NDUERR****Register type**

Peg

**Description**

NDUERR counts the number of errors on an in-service or out-of-service unit of a node. The register counts errors if the system performs additional action on these errors. The events counted range from one-time hits to total failures.

The register increases when an application on the node unit reports an error that is a result of either a manual or a system maintenance action.

**Associated registers**[NDUSWERR](#), [NDUTRAP](#)**Extension registers**

None

**Associated logs**

None

**NDUFLT****Register type**

Peg

**Description**

NDUFLT counts the errors that remain after diagnostics. The register increases when the first diagnostic attempt does not clear the error. Additional tests of the error condition do not increase the fault register.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUMBP****Register type**

Peg

**Description**

NDUMBP counts the number of times the unit becomes ManB.

**Associated registers**[NDUMBU](#)**Extension registers**

None

**Associated logs**

None

**NDUMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NUTMBU measures the length of time that a unit is ManB.

**Associated registers**[NDUMBP](#)

**Extension registers**

None

**Associated logs**

None

**NDUMCRST****Register type**

Peg

**Description**

NDUMCRST counts the number of cold restarts that occur on a unit of a node as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUMRRST****Register type**

Peg

**Description**

NDUMRRST counts the number of reload restarts that occur on a unit as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUMWRST****Register type**

Peg

**Description**

NDUMWRST counts the number of warm restarts that occur on a unit as a result of manual action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUNAP****Register type**

Peg

**Description**

NDUNAP counts the number of times the system isolates a unit from the DMS.

**Associated registers**

[NDUNAU](#)

**Extension registers**

None

**Associated logs**

None

**NDUNAU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NDUNAU measures the length of time the system isolates a unit from the DMS. The system takes a sample every 100 seconds.

**Associated registers**

[NDUNAP](#)

**Extension registers**

None

**Associated logs**

None

**NDUSBP****Register type**

Peg

**Description**

NDUSBP counts the number of times that the system puts a unit into the system busy (SYSB) state.

**Associated registers**

[NDUSBU](#)

**Extension registers**

None

**Associated logs**

None

**NDUSBU****Register type**

Usage

**Scan rate**

Not available

**Description**

NDUSBU measures the length of time a unit is SysB.

**Associated registers**

[NDUSBP](#)

**Extension registers**

None

**Associated logs**

None

**NDUSCRST****Register type**

Peg

**Description**

NDUSCRST counts the number of cold restarts that occur on a unit as a result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUSRRST****Register type**

Peg

**Description**

NDUSRRST counts the number of reload restarts that occur on a unit as a result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUSWERR****Register type**

Peg

**Description**

NDUSWERR counts the number of software errors that occur on a unit.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUSWRST****Register type**

Peg

**Description**

NDUSWRST counts the number of warm restarts that occur on a unit as a result of system action.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDUTRAP****Register type**

Peg

**Description**

NDUTRAP counts the number of traps that occur on a unit.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NPAPEG

### Description

OM group Numbering Plan Area (NPAPEG) counts provide the ability to determine bottlenecks in the network, and to determine where new trunk groups are required.

The following table lists the key and info fields associated with OM group NPAPEG:

| Key field | Info field |
|-----------|------------|
| NPAVALS   | None       |

### Related functional groups

There are no functional groups associated with OM group NPAPEG.

### Registers

The following table lists the registers associated with OM group NPAPEG and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NPAPEG

| Register name       | Measures        |
|---------------------|-----------------|
| <a href="#">ATB</a> | All Trunks Busy |

#### ATB

##### Register type

Peg

##### Description

ATB increments each time an all trunks busy (ATB) condition is reached on a route or choice. It may be pegged more than once per call if the second or successive choice of trunks is also busy. ATB pegs per assigned NPA.

##### Associated registers

None

##### Extension registers

None



**Associated logs**  
ATB100

## NRS

### Description

OM group Network Resource Selector (NRS) provides information on the use of network resource selectors (NRS). The modem pools (MP) are the only NRS type that the system monitors.

The following table lists the key and info fields associated with OM group NRS:

| Key field | Info field                                                   |
|-----------|--------------------------------------------------------------|
| CLLI      | Resource type. The only correct resource type for NRS is MP. |

### Related functional groups

There are no functional groups associated with OM group NRS.

### Registers

The following table lists the registers associated with OM group NRS and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NRS (Sheet 1 of 2)

| Register name           | Measures                                                     |
|-------------------------|--------------------------------------------------------------|
| <a href="#">NRSCON</a>  | Successful connections by network resource selectors         |
| <a href="#">NRSCONU</a> | Connected usage                                              |
| <a href="#">NRSFCON</a> | Not complete attempts to connect a network resource selector |
| <a href="#">NRSFRES</a> | Not complete attempts to reserve network resource selector   |
| <a href="#">NRSMBU</a>  | Manual busy usage                                            |
| <a href="#">NRSNMP</a>  | No modem pool prefix dial attempts                           |
| <a href="#">NRSOVFL</a> | Network resource selector overflow                           |

**Registers for OM group NRS (Sheet 2 of 2)**

| Register name           | Measures                                                 |
|-------------------------|----------------------------------------------------------|
| <a href="#">NRSRES</a>  | Successful attempts to reserve network resource selector |
| <a href="#">NRSRESU</a> | Reserved usage                                           |
| <a href="#">NRSSBU</a>  | System busy usage                                        |

**NRSCON****Register type**

Peg

**Description**

NRSCON counts successful connections made by the network resource selector.

An attempt to connect a network resource selector is successful if a modem pool connects when a call process sets up. The attempt is also successful if a maintenance modem pool connects to the modem pool under test.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSCONU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NRSCONU records if the units of a network resource selector group are correct.

Modem pools connect when a call is in progress. Maintenance modem pools (MMP) connect when the network connection between the MMP and the modem pool under test establishes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSFCON****Register type**

Peg

**Description**

NRSFCON counts attempts to connect a network resource selector that are not successful.

An attempt to connect a network resource selector is not successful if a modem pool fails to connect during a call setup. The attempt is also unsuccessful if a maintenance modem pool fails to connect to the modem pool under test.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSFRES****Register type**

Peg

**Description**

NRSFRES counts attempts to reserve a network resource selector that are not successful.

An attempt to reserve a network resource selector is not successful if a modem pool or maintenance modem pool is not correctly removed from the free queue.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NRSMBU records if units of a network resource selector group are manual busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSNMP****Register type**

Peg

**Description**

NRSNMP increases when a data unit in a network resource selector group dials the entry no modem pool (NMP) prefix. The data unit contains NRS default Outbound.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSOVFL****Register type**

Peg

**Description**

NRSOVFL increases when an NRS does not have any free units. The NRS overflows to another group to find a free unit.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSRES****Register type**

Peg

**Description**

NRSRES counts successful attempts to reserve a unit of a network resource selector.

An attempt to reserve a unit of a network resource selector is successful if the system removes a modem from the free queue. The system uses the modem for call processing or maintenance.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSRESU****Register type**

Usage

**Scan rate**

10 seconds

**Description**

NRSRESU records if units of a network resource selector group are reserved.

The system reserves modem pools and maintenance modem pools when the system takes these pools off the free queue.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NRSSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

NRSSBU records if units of a network resource selector group are system busy.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NSC

### Description

OM group Number Services Code Call Summary (NSC) provides summary information on number services code (NSC) calls. The NSC calls require access to service control point (SCP) databases. The OM group NSC indicates the grade of service provided by a service switching point (SSP).

The following table lists the key and info fields associated with OM group NSC:

| Key field                                   | Info field |
|---------------------------------------------|------------|
| Defines NSCORIG (NSC) code in table NSCDEFS | None       |

The TIMEOUT and OPTIONS information fields in NSCDEFS must contain NSC codes for NSCT2TO increases.

### Related functional groups

The following functional groups are associated with OM group NSC:

- 800+
- E800
- E008
- PVN
- Common Channel Signaling 7 (CCS7)

### Registers

The following table lists the registers associated with OM group NSC and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NSC (Sheet 1 of 2)

| Register name            | Measures                                                 |
|--------------------------|----------------------------------------------------------|
| <a href="#">NSCABNAS</a> | NSC call abandon before answered                         |
| <a href="#">NSCABNBS</a> | NSC call abandon before the seizure of an outgoing trunk |



**Registers for OM group NSC (Sheet 2 of 2)**

| Register name                               | Measures                                                            |
|---------------------------------------------|---------------------------------------------------------------------|
| <a href="#"><u>NSCATIN</u></a>              | NSC access tandem trunk incoming                                    |
| <a href="#"><u>NSCDBOVL-Canada only</u></a> | NSC database overload response                                      |
| <a href="#"><u>NSCEIGHT-Canada only</u></a> | NSC 800 number returned                                             |
| <a href="#"><u>NSCFLICM</u></a>             | NSC invalid command message                                         |
| <a href="#"><u>NSCFLICS</u></a>             | NSC invalid command sequence                                        |
| <a href="#"><u>NSCFPRIQ-Canada only</u></a> | NSC failure before query                                            |
| <a href="#"><u>NSCINVY-Canada only</u></a>  | NSC invalid special routing code (SRC)/00Y code                     |
| <a href="#"><u>NSCIVCAR</u></a>             | NSC call invalid carrier identification                             |
| <a href="#"><u>NSCNSNPA-Canada only</u></a> | NSC number of non-subscribed numbering plan area (NPA) responses    |
| <a href="#"><u>NSCORIG</u></a>              | NSC originated                                                      |
| <a href="#"><u>NSCOUTSV-Canada only</u></a> | NSC out-of-service responses                                        |
| <a href="#"><u>NSCQUERY-Canada only</u></a> | NSC query                                                           |
| <a href="#"><u>NSCSFLEA</u></a>             | NSC failure to receive second signaling stage on equal access trunk |
| <a href="#"><u>NSCSFLT0</u></a>             | NSC signaling failure timeout                                       |
| <a href="#"><u>NSCT2TO</u></a>              | NSC T2 timeout                                                      |
| <a href="#"><u>NSCTIOVF</u></a>             | NSC transaction identification not available before initial query   |
| <a href="#"><u>NSCUNSOR-Canada only</u></a> | NSC unsolicited responses                                           |
| <a href="#"><u>NSCVACDR-Canada only</u></a> | NSC vacant database responses                                       |

**NSCABNAS****Register type**

Peg

**Description**

NSCABNAS increases when the system receives an on-hook message from the calling party. The system receives this message after an SSP seizes an outgoing trunk and before the user answers the call.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCABNBS****Register type**

Peg

**Description**

NSCABNBS increases when the system receives an on-hook message from the calling party before an SSP seizes an outgoing trunk.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCATIN****Register type**

Peg

**Description**

NSCATIN counts NSC calls that the system receives from other offices (trunk calls) in:

- E800

NSCATIN counts NSC calls that the system receives from other offices (trunk calls) in an access tandem/SSP system. This count

increases for calls that originate from toll trunks. Examples of toll trunks are Intertoll, Supercama, and TOPS.

In an equal access end office (EAEO)/SSP system, the count is zero.

NSCATIN counts PVN calls on super-centralized automatic message accounting (Supercama) and inter-toll trunks. Register NSCATIN counts PVN calls even if the application of Automatic Call Gapping occurs.

- 800PLUS

NSCATIN counts NSC calls that the system receives from other offices (trunk calls) in a DMS-200 or DMS-100/200 system. This count increases for calls that originate from toll trunks. Examples of toll trunks are Intertoll, Supercama, and TOPS.

In a DMS-100 system, the count is zero.

- E008

In a DMS system that uses E008, the count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCDBOVL- Canada only****Register type**

Peg

**Description**

NSCDBOVL increases if a database returns the subsystem congestion diagnostic to an SSP. The return indicates a database overload.

The call routes to reorder (RODR) treatment.

When the system launches a query to a database using CCS7, part of the signaling-connection control part (SCCP) information specified in the query is an option field. If this option is set to RETURN TO ERROR, the database query launched by an SSP may return to the SSP by the SCP database when a routing failure occurs.

For all 800+ calls, the SCCP option is set to return a message to the database if an error occurs. Part of the message returned to the database is a diagnostic field. Possible values for the diagnostic field are subsystem failure, unequipped user, and subsystem congestion.

In a PVN, the NSCDBOVL count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCEIGHT- Canada only****Register type**

Peg

**Description**

NSCEIGHT increases when an 800 number returns from the SCP database. This action indicates that the 800 number does not switch over to the SSP for 800+. The number is translated again using the INWATS tables.

In a PVN system, the NSCEIGHT count is zero.

For E008 FPS, this register indicates the number of times that the response from the SCP contains the special routing parameter set to a movement number in a routing component returned from the SCP. The call continues based on non-E008 translations.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCFLICM****Register type**

Peg

**Description**

NSCFLICM increases when the SSP receives an undecipherable response from the SCP.

The call routes to reorder (RODR) treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCFLICS****Register type**

Peg

**Description**

NSCFLICS increases when the SSP receives a response from the SCP that contains not complete or not-in-sequence commands.

The call routes to reorder (RODR) treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCFPRIQ- Canada only****Register type**

Peg

**Description**

NSCFPRIQ counts 800+ calls that fail before a database query is launched, including calls that fail for one of the following reasons:

- invalid called number digits
- 800+ subsystem out of service (OOS)
- there are no transaction identifiers available

PVN calls that fail before a database query launches are also counted by NSCFPRIQ, including calls that fail for one of the following reasons:

- PVN subsystem OOS
- wrong PVN transaction identification
- invalid calling number digits
- global title cannot be formatted
- invalid local access and transport area (LATA) number digits
- invalid dial call type for PVN in encode data
- invalid digits in originating number on remote access call
- wrong number of digits in originating number remote access call
- invalid digits in authorization code or personal identification number (PIN)
- package encoding fails on transaction capabilities application part (TCAP) message

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCINVY- Canada only****Register type**

Peg

**Description**

NSCINVY counts invalid SRC/00Y codes received by the SSP from the end office. An end office may substitute an SRC/00Y code for the 800 code in the 800 + NXX + XXXX number to indicate the originating numbering plan area to the SSP. The code is considered invalid if it is not entered in table NSCSNPA.

The call routes to vacant code treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NSC100

**NSCIVCAR****Register type**

Peg

**Description**

NSCIVCAR increases when the database returns an invalid carrier identification in the response message. A carrier identification is invalid if it is not datafilled in a correct office table of correct carrier identifiers for the number service call service.

The call routes to CCS7 application failure treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NSC100

**NSCNSNPA- Canada only****Register type**

Peg

**Description**

NSCNSNPA increases if the database returns out of band (out of zone) as a special routing in the database response.

Use this register for the 800+ feature. In an SSP E800 office and in a PVN, the NSCNSNPA count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCORIG****Register type**

Peg

**Description**

NSCORIG counts NSC line calls that reach the dial complete stage. NSC includes all NSC calls that originate from lines, attendant consoles, and PX type trunks. In addition, all NSC calls that result from call redirection (e.g. call forwarding, ACD/UCD on night service to 800, etc.) are included in this OM.

- E800

In an EAEO/SSP system, the count is the total number of NSC calls from lines. This count includes NSC calls that result from call redirection. In a DMS-100/200 combined access tandem/SSP office, the count is the total number of NSC calls originated by collocated stations (line calls) plus NSC calls that result from call redirection. In a DMS-200 access tandem/SSP office, the count is zero.

PVN calls on integrated business network (IBN) trunks and lines, consoles, IBNT1 trunks, and IBNT2 trunks are counted by this register.

- 800PLUS

In a DMS-100 end office, SSP or DMS-100/200 SSP, the count is the total number of NSC calls from lines or collocated stations (line calls) plus NSC calls that result from call redirection. In a DMS-200, the count is zero.

PVN calls on integrated business network (IBN) trunks and lines, consoles, IBNT1 trunks, and IBNT2 trunks are counted by this register.

- E008

In a DMS-100 end office, SSP or DMS-100/200 SSP, the count is the total number of NSC calls originated by lines or collocated stations (line calls) plus NSC calls that result from call redirection plus calls originated over AISUP and ATUP (IBN type) trunks. In a DMS-200, the count is zero.

**Associated registers**

None

**Extension registers**

None



**Associated logs**

None

**NSCOUTSV- Canada only****Register type**

Peg

**Description**

NSC out-of-service responses (NSCOUTSV) increases if a database returns the subsystem failure diagnostic and indicates that the database is not available.

The call routes to reorder (RODR) treatment.

When a query launches to a database using CCS7, part of the signaling connection control part (SCCP) information specified in the query is an option field. If this option is set to RETURN ON ERROR, the database query launched by an SSP may return to the SSP by the SCP database when a routing failure occurs.

For all 800+ calls, the SCCP option is set to RETURN ON ERROR. Part of the message returned to the database is a diagnostic field. Possible values for the diagnostic field are subsystem failure, unequipped user, and subsystem congestion.

In a PVN, the NSCOUTSV count is zero.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCQUERY- Canada only****Register type**

Peg

**Description**

NSCQUERY that counts 800 database queries and are sent by call processing uses the transaction capabilities application part (TCAP). This register also counts database queries required for PVN calls.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCSFLEA****Register type**

Peg

**Description**

NSCSFLEA increases when the first stage of signaling (KP + OZZ + XXX + ST) from the EAEO indicates an NSC call or a PVN call, but either no second-stage signaling is received or the second stage is incomplete.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCSFLTO****Register type**

Peg

**Description**

NSCSFLTO increases when a reply is not sent back to the SSP from the SCP within the time interval specified in table NSCDEFS.

The call routes to reorder (RODR) treatment.

This register also applies to calls on the PVN.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCT2TO****Register type**

Peg

**Description**

NSCT2TO increases when the SSP sends a query to the SCP and does not receive a response message from the SCP within the T2 time interval specified in table NSCDEFS.

The T2 time interval is an optional parameter of table NSCDEFS. NSCT2TO will always be zero for keys where the corresponding tuples in table NSDEFS are not entered with the T2 timeout option.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCTIOVF****Register type**

Peg

**Description**

NSCTIOVF increases when an SSP NSC call fails because the NSC transaction identification is not available in the SSP. This register also applies to PVN calls.

The call routes to reorder (RODR) treatment.

The office parameter uses NO\_OF\_TRANSACTION\_IDS in table OFCENG to allocate the number of transaction identifiers available to the SSP for launching database queries to an SCP database.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCUNSOR- Canada only****Register type**

Peg

**Description**

NSCUNSOR counts not requested responses that are received by an SSP from an SCP. Not requested responses from the database do not have a corresponding query.

Examples of not requested responses are those in which

- the transaction identification in the response is out of range (greater than the maximum number of queries)
- the transaction identification does not have a corresponding call connected to it
- a response to a database query returns after the database query has timed out

It is not correct to wait for every response beyond the timeout period, because all the transaction identifiers could be lost during the wait. The database timeout value should be set so that all normal responses (that is, the responses not involving database or network problems) can be received from the database within the timeout period. This register will generally indicate how many responses take too long.

NSCUNSOR counts false responses and late responses. The two are not distinguishable. This register also applies to the PVN system.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCVACDR- Canada only****Register type**

Peg

**Description**

NSCVACDR increases if the database response indicates a vacant code. In a PVN system, the NSCVACDR count is zero.

The call is routes to vacant code (VACT) treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NSCACG

### Description

OM group Number Services Code Automatic Call Gapping (NSCACG) provides information on the performance of automatic call gapping (ACG) for number services code (NSC) calls. The Service Management System (SMS) uses ACG to implement network management controls.

The following table lists the key and info fields associated with OM group NSCACG:

| Key field | Info field |
|-----------|------------|
| NSC_INDEX | None       |

### Related functional groups

The following functional groups associated with OM group NSCACG:

- Common Channel Signaling 7 (CCS7)
- 800 Plus (800+) Service
- E800 Service
- PVN

### Registers

The following table lists the registers associated with OM group NSCACG and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NSCACG (Sheet 1 of 2)

| Register name            | Measures                                 |
|--------------------------|------------------------------------------|
| <a href="#">NSCATMPT</a> | NSC attempts                             |
| <a href="#">NSCBKMCC</a> | NSC blocked mass calling controls        |
| <a href="#">NSCBKSIC</a> | NSC blocked by Service Management System |
| <a href="#">NSCBKSOC</a> | NSC blocked SCP overload controls        |
| <a href="#">NSCBKVC</a>  | NSC blocked vacant (VACT) codes          |
| <a href="#">NSCCOMC</a>  | NSC mass calling control list overflow   |

**Registers for OM group NSCACG (Sheet 2 of 2)**

| Register name            | Measures                                                            |
|--------------------------|---------------------------------------------------------------------|
| <a href="#">NSCCONPN</a> | NSC non-purchased NPA control list overflow                         |
| <a href="#">NSCCOSCP</a> | NSC service control point (SCP) control list overflow               |
| <a href="#">NSCCOSI</a>  | NSC service management system (SMS)-initiated control list overflow |
| <a href="#">NSCCOSVC</a> | NSC six-digit vacant (VACT) code control list overflow              |
| <a href="#">NSCCOTVC</a> | NSC ten-digit vacant (VACT) code control list overflow              |

**NSCATMPT****Register type**

Peg

**Description**

NSCATMPT counts line and trunk originating E800 calls that reach the SSP. The calls contain OM Registers: NSC\_NSCORIG, NSC\_NSCATIN, NSCACG\_NSCBKVC, NSCACG\_NSCBKSOC, NSCACG\_NSCBKMCC, and NSCACG\_NSCBKSIC.

**Associated registers**

NSXACG\_NSCATMPT, NSC\_NSCORIG, NSC\_NSCATIN, [NSCBKVC](#), [NSCBKSOC](#), [NSCBKMCC](#), [NSCBKSIC](#)

**Extension registers**

None

**Associated logs**

None

**NSCBKMCC****Register type**

Peg

**Description**

NSCBKMCC counts NSC calls that ACG controls for ten-digit mass calling controls block.

The system routes NSC calls blocked for mass calling controls to busy line (BUSY) treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCBKSIC****Register type**

Peg

**Description**

NSCBKSIC counts NSC calls that ACG controls block. The SMS initiates ACGs and forwards them through a service control point to the correct service switching point.

The system routes the NSC calls that ACG-initiated controls block to reorder (RODR) treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCBKSOC****Register type**

Peg

**Description**

NSCBKSOC counts NSC calls that ACG controls for SCP overloads block.

NSC calls blocked by SCP overload controls route to general no circuit (GNCT) treatment.

**Associated registers**

None



**Extension registers**

None

**Associated logs**

None

**NSCBKVC****Register type**

Peg

**Description**

NSCBKVC counts calls that ACG controls block. The system applies ACG controls when one of the following occurs: VACT codes receive too many calls, or too many calls are made from numbering plan areas (NPA) that are not purchased for NSCs.

The NSC calls blocked for greater than necessary calling to VACT codes that the system routes to VACT code treatment. The system routes NSC calls blocked for greater than necessary calling. The systems routes the calls that come from non-purchased NPAs and go to not authorized INWATS (UNIN) call treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOMC****Register type**

Peg

**Description**

NSCCOMC increases when an ACG control cannot apply to a code for an 800 number because the control list is full.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCONPN****Register type**

Peg

**Description**

NSCCONPN increases when an ACG control on a code is not placed because the control list for calls is full. The calls come from NPAs that are not purchased for NSC use.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOSCP****Register type**

Peg

**Description**

NSCCOSCP increases when a required ACG control that SCP overloads is not placed on a code. The SCP overload is not placed on a code because the control list is full.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOSCP****Register type**

Peg

**Description**

NSCCOSCP increases when a required ACG control that SCP overloads is not placed on a code. The SCP overload is not placed on a code because the control list is full.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOSI****Register type**

Peg

**Description**

NSCCOSI increases when an ACG control that the SMS initiates is not placed on a code because the control list is full.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOSVC****Register type**

Peg

**Description**

NSCCOSVC increases when an ACG control is not placed on a VACT six-digit code. The ACG control is not placed on a code because the control list for six-digit codes is full. The ACG control is also not placed on a code because the control list is full.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NSCCOTVC****Register type**

Peg

**Description**

NSCCOTVC increases when an ACG control is not placed on a VACT ten-digit code. An ACG is not placed on a code because the control list for ten-digit codes is full.

The DMS-100 can control a maximum of 64 ten-digit NSCs and 256 six-digit NSCs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NWMFRRCT

### Description

OM group Network Management Flexible Reroute (NWMFRRCT) counts calls that are rerouted, and rerouted calls that fail to find an idle VIA route. The counts are made for each switch.

Calls are rerouted from an in-chain route to a VIA route. In-chain routes are trunk groups that carry calls according to the rules for routing in a hierarchical network. VIA routes are trunk groups that carry rerouted calls for which the network routing rules for the hierarchical network are ignored.

The following table lists the key and info fields associated with OM group NWMFRRCT.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group NWMFRRCT.

### Registers

The following table lists the registers associated with OM group NWMFRRCT and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NWMFRRCT

| Register name            | Measures                          |
|--------------------------|-----------------------------------|
| <a href="#">FRRATTCT</a> | Flexible reroutes attempt control |
| <a href="#">FRRFLCT</a>  | Flexible reroutes failed control  |

**FRRATTCT****Register type**

Peg

**Description**

FRRATTCT counts calls that are rerouted to a VIA route list.

**Associated registers**

None

**Extension registers**

FRRATTC2

**Associated logs**

None

**FRRFLCT****Register type**

Peg

**Description**

FRRFLCT counts rerouted calls that fail to find an idle VIA route list.

**Associated registers**

None

**Extension registers**

FRRFLCT2

**Associated logs**

None

## NWMSILC

### Description

OM group Network Management Selective Incoming Load Control (NWMSILC) counts calls that the network management selective-incoming load control (SILC) blocks.

The SILC permits incoming and two-way trunk groups to limit incoming calls according to preset rate, percentage values, or both. The preset rate and percentage value are in Table NWMIDOC.

When the SILC blocks a call, the system sends a start dial signal to permit the far-end sender to out-pulse digits. The system ignores the digits and connects a tone in the peripheral module to warn the caller that the call failed. This action makes the trunk available for normal call processing after the caller disconnects.

The following table lists the key and info fields associated with OM group NWMSILC:

| Key field                                                                          | Info field |
|------------------------------------------------------------------------------------|------------|
| CLLI for the trunk group. The CLLI is the external identifier for the trunk group. | None       |

### Related functional groups

There are no functional groups associated with OM group NWMSILC.

### Registers

The following table lists the registers associated with OM group NWMSILC and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group MWMSILC

| Register name           | Measures                              |
|-------------------------|---------------------------------------|
| <a href="#">TRKSILC</a> | Trunk selective incoming load control |

#### TRKSILC

**Register type**  
Peg

**Description**

TRKSILC increases when the selective incoming load control blocks a trunk group. Load controls block a trunk group when the trunk group receives too many calls.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None



---

## NWMTGCNT

---

### Description

OM group Network Management Trunk Group Control (NWMTGCNT) counts calls that are encountered and affected by each type of network management trunk group (NWM TG) control.

NWM TG controls are classified as either expansive or protective. Expansive trunk group controls modify the available routes a call can take, thereby increasing the likelihood that a call will reach its proper destination when a network is congested. Protective trunk group controls protect the network when it is congested by preventing calls from entering the network.

The following are examples of protective trunk groups and their purpose:

- Directional reservation (DRE) gives priority to incoming calls on a controlled trunk group, rather than outgoing calls.
- Protective reservation (PRE) gives priority to direct routed calls offered to a controlled trunk group.
- Cancel-to (CANT) blocks calls that access a controlled trunk group.
- SKIP prevents calls from being offered to a controlled trunk group, thereby causing those calls to advance to the next trunk group in a route list.
- Cancel-from (CANT) blocks calls that overflow a controlled trunk group.
- Incoming trunk busy (ITB) restricts the number of incoming calls on a controlled trunk group that has the remote-make-busy capability (assigned in table TRKSGRP). This control removes from service a percentage of the trunks in a trunk group if the number of idle trunks falls below a predefined threshold.
- Selective trunk reservation (STR) blocks outgoing calls if the number of idle trunks in a trunk group falls below a predefined threshold.
- Bidirectional trunk group reservation control (BRC) blocks outgoing calls under the following condition: the number of idle trunks falls below the number of trunks reserved for incoming calls, the number of outgoing calls is greater than or equal to the number of trunks reserved for outgoing calls, and the number of priority calls is greater than or equal to the number of trunks reserved for priority calls.

The following table lists the key and info fields associated with OM group NWMTGCNT:

| Key field                                                                        | Info field |
|----------------------------------------------------------------------------------|------------|
| NWM_GRP_CONTROL. The names of the NWM TG controls make up the key to this group. | None       |

## Related functional groups

There are no functional groups associated with OM group NWMTGCNT.

## Registers

The following table lists the registers associated with OM group NWMTGCNT and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group NWMTGCNT

| Register name            | Measures                                        |
|--------------------------|-------------------------------------------------|
| <a href="#">NWMTGAFF</a> | NWM TG affected                                 |
| <a href="#">NWMTGATT</a> | NWM TG attempts                                 |
| <a href="#">BSSKIP</a>   | Bearer Service Skip                             |
| <a href="#">BSSNSPCH</a> | BSS No capacity or speech                       |
| <a href="#">BSSN3K1</a>  | Register BSS No channels available for 3.1kHz   |
| <a href="#">BSSN64K</a>  | BSS No 64kbit/s unrestricted capacity available |
| <a href="#">BSSNSPCU</a> | BSS No capacity for speech                      |
| <a href="#">BSSN3K1U</a> | BSS No channels available for 3.1kHz            |
| <a href="#">BSSN64KU</a> | BSS No channels available for 3.1kHz            |

### NWMTGAFF

#### Register type

Peg

**Description**

NWMTGAFF counts calls that are directly affected by an NWM TG control. Depending on the control type, affected calls may be blocked, or may skip to the next trunk group in the route list.

ITB messages cannot be recorded. The entry corresponding to ITB is always zero.

NWMTGAFF increases when BRC prevents a call from accessing the trunk group to which it is routed.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NWMTGATT****Register type**

Peg

**Description**

NWMTGATT counts calls that encounter the NWMTG control type.

ITB messages cannot be recorded. The entry corresponding to ITB is always zero.

NWMTGATT increases when BRC is active on a trunk group selected for an outgoing call.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**BSSKIP****Register type**

Peg

**Description**

BSSKIP measures the number of trunk groups that have been skipped over during routing procedure, because they have had BSSKIP control active.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**BSSNSPCH****Register type**

Peg

**Description**

BSSNSPCH counts how many No capacity for speech signals have been received from DCME.

- PEG: Y
- USAGE: N
- HIGH WATER:N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

DCME105

**BSSN3K1****Register type**

Peg

**Description**

BSSN3K1 counts how many no channels available for 3.1 kHz signals have been received from DCME.

- PEG: Y
- USAGE: N

- HIGH WATER: N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

DCME105

**BSSN64K****Register type**

Peg

**Description**

This usage count is accumulated as a result of a 10 second scan of 'No 64kbit/s unrestricted capacity available' signal.

- PEG: N
- USAGE: Y
- HIGH WATER: N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

DCME105

**BSSNSPCU****Register type**

Peg

**Description**

This usage count is accumulated as a result of a 10 second scan of 'No capacity for speech' signal.

- PEG: N
- USAGE: Y

- HIGH WATER: N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**BSSN3K1U****Register type**

Peg

**Description**

This usage count is accumulated as a result of a 10 sec scan of 'No channels available for 3.1kHz' signal.

- PEG: Y
- USAGE: N
- HIGH WATER: N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

DCME105

**BSSN64KU****Register type**

Peg

**Description**

This usage count is accumulated as a result of a 10 sec scan of 'No 64kbit/s unrestricted capacity available' signal.

- PEG: N
- USAGE: Y

- HIGH WATER: N
- OTHER: N/A

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NX25L2

### Description

OM group NX25 Level 2 (NX25L2) contains the NX25 Level 2 OMs that refer to the data links.

The following table lists the key and info fields associated with OM group NX25L2:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group NX25L2.

### Registers

The following table lists the registers associated with OM group NX25L2 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NX25L2

| Register name            | Measures                   |
|--------------------------|----------------------------|
| <a href="#">NLINKUP</a>  | NX25 Link Up               |
| <a href="#">NLINKDN</a>  | NX25 Link Down             |
| <a href="#">NBSYNTRN</a> | NX25 Busy Not Transferred  |
| <a href="#">NBSYNACK</a> | NX25 Busy Not Acknowledged |

#### NLINKUP

##### Register type

Peg

##### Description

NLINKUP increments when an NX25 link is successfully brought into service.

##### Associated registers

None



**Extension registers**

None

**Associated logs**

NPAC125

**NLINKDN****Register type**

Peg

**Description**

NLINKDN increments when an NX25 link is brought down.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC126

**NBSYNTRN****Register type**

Peg

**Description**

NBSYNTRN increments when a packet is refused because there are currently seven untransferred packets.

**Note:** A packet is a group of binary digits including data and call control signals processed as a composite whole. The data, call control signals, and possible error control information are arranged in a specified format.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NBSYNACK****Register type**

Peg

**Description**

NBSYNACK increments when a packet is refused because the last packet has not been acknowledged, and the timer has not expired.

**Note:** A packet is a group of binary digits including data and call control signals processed as a composite whole. The data, call control signals, and possible error control information are arranged in a specified format.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## NX25L3

### Description

OM group NX25 Level 3 (NX25L3) contains the OM s that refer to the virtual channel identifiers (VCI) on the 6X91BA card.

The following table lists the key and info fields associated with OM group NX25L3:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group NX25L3.

### Registers

The following table lists the registers associated with OM group NX25L3 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NX25L3

| Register name            | Measures                                              |
|--------------------------|-------------------------------------------------------|
| <a href="#">NVCUIP</a>   | NX25 Virtual Channel Identifier Up                    |
| <a href="#">NVCIFAIL</a> | NX25 Virtual Channel Identifier Initialization Failed |
| <a href="#">NVCINRMT</a> | NX25 Virtual Channel Identifier Normal Termination    |
| <a href="#">NVCIAVRT</a> | NX25 Virtual Channel Identifier Aborted               |

#### NVCUIP

##### Register type

Peg

##### Description

NVCUIP increments when a VCI is successfully brought into service.

##### Associated registers

None

**Extension registers**

None

**Associated logs**

NPAC110

**NVCIFAIL****Register type**

Peg

**Description**

NVCIFAIL increments when a VCI initialization fails.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC115

**NVCINRMT****Register type**

Peg

**Description**

NVCINRMT increments when a VCI terminates normally.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC111

**NVCIABRT****Register type**

Peg

**Description**

NVCIABRT increments when a VCI is aborted.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC116

## NX25MLP

### Description

OM group NX25 Multilink Procedures (NX25MLP) contains the multilink procedure (MLP)-based OMs. These OMs are pegged for the maximum number of MLP groups. Each multilink group has its own set of registers that are pegged once each time an event occurs.

The following table lists the key and info fields associated with OM group NX25MLP:

| Key field | Info field |
|-----------|------------|
| MLGID     | None       |

### Related functional groups

There are no functional groups associated with OM group NX25MLP.

### Registers

The following table lists the registers associated with OM group NX25MLP and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group NX25MLP

| Register name            | Measures                        |
|--------------------------|---------------------------------|
| <a href="#">MLINKUP</a>  | Multilink Up                    |
| <a href="#">MLINKDN</a>  | Multilink Down                  |
| <a href="#">MGRPUP</a>   | Multilink Group Up              |
| <a href="#">MGRPDN</a>   | Multilink Group Down            |
| <a href="#">MFRMRX</a>   | Multilink Frame Retransmissions |
| <a href="#">MTIMEOUT</a> | Multilink Timeout               |
| <a href="#">MFRMLOST</a> | Multilink Procedure Frame Lost  |
| <a href="#">MWINDERR</a> | Multilink Window Error          |

**MLINKUP****Register type**

Peg

**Description**

MLINKUP increments when a link in an MLP group is brought up.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC200

**MLINKDN****Register type**

Peg

**Description**

MLINKDN increments when a link in the MLP group fails.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NPAC201

**MGRPUP****Register type**

Peg

**Description**

MGRPUP increments when an entire MLP group is brought into service.

**Associated registers**

None

**Extension registers**

None

**Associated logs**  
NPAC201

## **MGRPDN**

**Register type**  
Peg

**Description**  
MGRPDN increments when an entire MLP group fails and is put out of service.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
NPAC202

## **MFRMRX**

**Register type**  
Peg

**Description**  
MFRMRX is the number of MLP frame retransmissions.

**Note:** A frame is one complete cycle of events in time division multiplexing. The frame usually includes a sequence of time slots for the various subchannels and extra bits for control.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
None

## **MTIMEOUT**

**Register type**  
Peg



**Description**

MTIMEOUT register increments when the lost frame timer expires on a lost frame.

**Note:** A frame is one complete cycle of events in time division multiplexing. The frame usually includes a sequence of time slots for the various subchannels and extra bits for control.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MFRMLOST****Register type**

Peg

**Description**

MFRMLOST increments when an incoming MLP frame is lost.

**Note:** A frame is one complete cycle of events in time division multiplexing. The frame usually includes a sequence of time slots for the various subchannels and extra bits for control.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MWINDERR****Register type**

Peg

**Description**

MWINDERR increments when an MLP window error occurs.

**Note:** A frame is one complete cycle of events in time division multiplexing. The frame usually includes a sequence of time slots for the various subchannels and extra bits for control.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## OADATCOM

### Description

Operator Services System Advanced Intelligent Network (OSSAIN) Data Communications

OADATCOM (OSSAIN Data Communications) is created for data communications operational measurements. The following OM groups are also created for data communications operational measurements:

- OANODEDC - OSSAIN Node Data Communications
- OASNPLDC - OSSAIN Session Pool Data Communications

OM group OADATCOM provides peg counts for OSSAIN data communications messaging events. It provides counts for the total number of messages sent from the CM to other nodes and the total number of messages received by the CM from other nodes. Counts of messages are broken down into successful counts and failure counts.

OM group OADATCOM provides two tuples for each office.

| Key field | Info field |
|-----------|------------|
| 0 to 1    | UDP or TCP |

The Info field value UDP or TCP is associated with the protocol used by the OSSAIN application. Currently call processing and maintenance use UDP, and QMS MIS uses TCP.

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OADATCOM. In release NA009, the group is changed to OSSAIN (OSAN0001).

## Registers

The following table lists the registers associated with OM group OADATCOM and what they measure. For a description of a register, click on the register name.

### Registers for OM group OADATCOM

| Register name            | Measures                             |
|--------------------------|--------------------------------------|
| <a href="#">OMSGRCFL</a> | OSSAIN message receive failure       |
| <a href="#">OMSGRCSC</a> | OSSAIN message receive success       |
| <a href="#">OMSGRCV</a>  | OSSAIN message received              |
| <a href="#">OMSGSND</a>  | OSSAIN message send requested        |
| <a href="#">OMSGSNFL</a> | OSSAIN message send failure          |
| <a href="#">OMSGSNSC</a> | OSSAIN message send success          |
| <a href="#">ORCVRTFL</a> | OSSAIN message receive route failure |
| <a href="#">OSNDRTFL</a> | OSSAIN message send route failure    |

### OMSGRCFL

#### Register type

Peg

#### Description

OSSAIN message receive failure

This register is pegged each time data communications encounters an error while attempting to forward an external node originated message to the destination DMS process. This can be caused by a failure in the DMS switch internal messaging system or data transport interface. This register is pegged also during failures indicated by register ORCVRTFL.

**Note:** Note: This register can be validated by summing the receive failed counts, register ONMSGRFL, of each node datafilled in table OANODINV.

$\text{OMSGRCFL} = \text{OMSGRCV} - \text{OMSGRCSC}$

$\text{OMSGRCFL} \geq \text{ORCVRTFL}$

This register is related to OM group OANODEDC register ONMSGRFL as follows: OMSGRCFL is equal or greater than the value of

ONMSGRFL for each node, added over all nodes in table OANODINV. This calculation is represented also as follows:

$$\text{OM group OADATCOM} \quad \text{OM group OANODEDC}$$

$$\text{OMSGRCFL} \geq \sum_{0}^n \text{ONMSGRFL}_n$$

where n = number of nodes datafilled in table OANODINV

#### Associated registers

[OMSGRCV](#), [OMSGRCSC](#), [ORCVRTFL](#), and ONMSGRFL

#### Extension registers

There are no extension registers.

#### Associated logs

Log numbers: OAIN605 and OAIN606

### OMSGRCSC

#### Register type

Peg

#### Description

OSSAIN message receive success

This register is pegged when the CM's data communications software is able to successfully process an incoming message.

**Note:** Note: This register can be validated by summing the successful message receives, register ONMSGRSC, of each node datafilled in table OANODINV.

$$\text{OMSGRCSC} = \text{OMSGRCV} - \text{OMSGRCFL}$$

This register is related to OM group OANODEDC as follows:

|                                                                                                                                                                                                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> OM group<br/> <u>OADATCOM</u> </div> <div style="text-align: center;"> OM group<br/> <u>OANODEDC</u> </div> </div> $\text{OMSGRCSC} \geq \sum_{n=0}^n \text{ONMSGRSC}_n$ <p style="text-align: center;">where n = number of nodes datafilled in table OANODINV</p> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Associated registers**

[OMSGRCV](#), [OMSGRCFL](#), and ONMSGRSC

**Extension registers**

OMSGRCS2

**Associated logs**

There are no associated logs.

**OMSGRCV****Register type**

Peg

**Description**

OSSAIN message received

This register is pegged for a specific node each time an incoming message, originating from an external node, is received from that node. This includes both call processing and maintenance messages.

**Note:** This register can be validated on a per node basis by adding the message receive success register and the message receive failure register that apply to the node of interest.

$$\text{OMSGRCV} = \text{OMSGRCSC} + \text{OMSGRCFL}$$

This register is related to OM group OANODEDC as follows:

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <u>OM group</u><br/> <u>OADATCOM</u> </div> <div style="text-align: center;"> <u>OM group</u><br/> <u>OANODEDC</u> </div> </div> <div style="text-align: center; margin-top: 20px;"> <math display="block">\text{OMSGRCV} \geq \sum_{0}^n \text{ONMSGRC}_n</math> </div> <div style="text-align: center; margin-top: 20px;"> <p>where n = number of nodes datafilled in table OANODINV</p> </div> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Associated registers**

[OMSGRCSC](#), [OMSGRCFL](#), and ONMSGRC

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**OMSGSND****Register type**

Peg

**Description**

OSSAIN message send requested

This register is pegged each time the data communications software is requested to send a message. This includes requests from call processing, maintenance, and Ethernet based QMS MIS messages.

**Note:** This register can be validated by adding the message send success register and the message send failure register.

$$\text{OMSGSND} = \text{OMSGSNSC} + \text{OMSGSNFL}$$

This register is related to OM group OANODEDC as follows:

|                                                                                                                                                                                                                                                                                                                                                              |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> OM group<br/> <u>OADATCOM</u> </div> <div style="text-align: center;"> OM group<br/> <u>OANODEDC</u> </div> </div> $\text{OMSGSND} \geq \sum_{0}^n \text{ONMSGSN}_n$ <p style="text-align: center;">where n = number of nodes datafilled in table OANODINV</p> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Associated registers**

[OMSGSNSC](#), [OMSGSNFL](#), and [OMSGSND](#)

**Extension registers**

OMSGSND2

**Associated logs**

There are no associated logs.

**OMSGSNFL****Register type**

Peg

**Description**

OSSAIN message send failure

This register is pegged each time data communications encounters an error while attempting to send an outgoing message. This can be caused by a data transport layer failure. This register is pegged also for reasons indicated by register OMSGSRFTL.

**Note:** This register can be validated by summing the failed message sends, register ONMSGSF, of each node datafilled in table OANODINV.

$\text{OMSGSNFL} = \text{OMSGSND} - \text{OMSGSNSC}$   
 $\text{OMSGSNFL} \geq \text{OSNDRFTL}$

This register is related to OM group OANODEDC as follows:



|                                                                               |                                                                               |
|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| <div style="text-align: center;"> <u>OM group</u><br/> <u>OADATCOM</u> </div> | <div style="text-align: center;"> <u>OM group</u><br/> <u>OANODEDC</u> </div> |
| $\text{OMSGSNFL} \geq \sum_{0}^n \text{ONMSGSL}_n$                            |                                                                               |
| <p>where n = number of nodes datafilled in table OANODINV</p>                 |                                                                               |

**Associated registers**

[OMSGSND](#), [OMSGSNSC](#), [OSNDRFTL](#), and ONMSGSL

**Extension registers**

There are no extension registers.

**Associated logs**

Log number: OAIN607

**OMSGSNSC****Register type**

Peg

**Description**

OSSAIN message send success

This register is pegged when the CM's data communications software is able to successfully process an outgoing message. Note that call processing and maintenance under OSSAIN uses non-guaranteed messaging, while QMS MIS uses TCP for guaranteed messaging. Pegging this register does not indicate that the message actually arrived at the destination node.

**Note:** This register can be validated by summing the successful message sends, register ONMSGSSC, of each node datafilled in table OANODINV.

$$\text{OMSGSNSC} = \text{OMSGSND} - \text{OMSGSNFL}$$

This register is related to OM group OANODEDC as follows:

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <u>OM group</u><br/> <u>OADATCOM</u> </div> <div style="text-align: center;"> <u>OM group</u><br/> <u>OANODEDC</u> </div> </div> <div style="text-align: center; margin-top: 20px;"> <math display="block">\text{OMSGSNSC} \geq \sum_{0}^n \text{ONMSGSC}_n</math> </div> <div style="text-align: center; margin-top: 20px;"> <p>where n = number of nodes datafilled in table OANODINV</p> </div> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Associated registers**

[OMSGSND](#), [OMSGSNFL](#), and ONMSGSSC

**Extension registers**

OMSGSNS2

**Associated logs**

There are no associated logs.

**ORCVRTFL****Register type**

Peg

**Description**

OSSAIN message receive route failure

This register is pegged each time the data communications software is unable to determine the destination of an external node originated message. This can be caused by a variety of reasons including:

- invalid protocol version
- invalid class header identifier
- invalid operation offset
- invalid message length
- invalid node identifier
- invalid session pool identifier
- invalid session identifier
- invalid network address

- invalid session pool state
- invalid node pool state
- invalid message size
- pool / node identifier mis-match
- corrupted message

**Note:** This register can be validated by summing the receive route failed counts, register ONRCRTFL, of each node datafilled in table OANODINV.

$$\text{ORCVRTFL} \leq \text{OMSGRCFL}$$

This register is related to OM group OANODEDC as follows:

$$\text{ORCVRTFL} \geq \sum_{n=0}^n \text{ONRCRTFL}_n$$

where n = number of nodes datafilled in table OANODINV

#### Associated registers

[OMSGRCFL](#) and ONRCRTFL

#### Extension registers

There are no extension registers.

#### Associated logs

Log number: OAIN605 and OAIN606

### OSNDRTFL

#### Register type

Peg

#### Description

OSSAIN message send route failure

This register is pegged each time the data communications software is unable to determine the destination of an outgoing message. This can be caused by the following reasons.

- invalid node identifier
- invalid session pool identifier
- invalid session identifier
- pool / node identifier mis-match
- corrupted message

**Note:** This register can be validated by summing the message send route failures, register ONSNRTFL, of each node datafilled in table OANODINV.

$$\text{OSNDRTFL} \leq \text{OMSGSNFL}$$

This register is related to OM group OANODEDC as follows:

|                                                        |                                    |  |
|--------------------------------------------------------|------------------------------------|--|
| <u>OM group</u><br><u>OADATCOM</u>                     | <u>OM group</u><br><u>OANODEDC</u> |  |
| $\text{OSNDRTFL} \geq \sum_{0}^n \text{ONSRTFL}_n$     |                                    |  |
| where n = number of nodes datafilled in table OANODINV |                                    |  |

#### Associated registers

[OMSGSNFL](#) and ONSNRTFL

#### Extension registers

There are no extension registers.

#### Associated logs

There are no associated logs.

## O AFLTRIG

### Description

OM group Operator Services System Advanced Intelligent Network (OSSAIN) Float Triggers (O AFLTRIG) provides peg counts for actions related to OSSAIN float trigger processing. These measurements cover float trigger processing causing call control to transfer to an OSSAIN function or control list.

The OSSAIN call float trigger tables (OACNNPRF, OADSCPRF, OATLKPRF, OACAUPRF, and OADTFPRF) must be datafilled with trigger events and actions, and a datafilled trigger event must occur for these registers to be pegged.

The following table lists the key and info fields associated with OM group O AFLTRIG:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group O AFLTRIG.

### Registers

The following table lists the registers associated with OM group O AFLTRIG and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group O AFLTRIG

| Register name             | Measures                       |
|---------------------------|--------------------------------|
| <a href="#">O AFLTCTL</a> | OSSAIN Float to a Control List |
| <a href="#">O AFLTFUN</a> | OSSAIN Float to a Function     |
| <a href="#">O ATRIGFL</a> | OSSAIN Trigger Failure         |

#### O AFLTCTL

**Register type**

Peg

**Description**

Calls in the floated state that trigger causing call control to be successfully passed to an OSSAIN Control List (in table OACTLDEF).

**Note:** For test case(s), make an OSSAIN call that performs float trigger processing to an OSSAIN Control List.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**OAFUTFUN****Register type**

Peg

**Description**

Calls in the floated state that triggered causing call control to be successfully passed to an OSSAIN Function (in table OAFUNDEF).

**Note:** For test case(s), make an OSSAIN call that performs float trigger processing to an OSSAIN Function.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**OATRIGFL****Register type**

Peg

**Description**

OATRIGFL indicates a trigger failure due to calls in the floated state that trigger and attempt to pass control to a Traffic Operator Position System (TOPS) automated system. Passing call control to a TOPS automated system via trigger processing is not allowed. This can occur as a result of 1) passing control to an OSSAIN Function which is a TOPS

automated system or 2) passing control to an OSSAIN Control List in which the first OSSAIN Function is a TOPS automated system.

**Note:** For test case(s), make an OSSAIN call that performs float trigger processing to an OSSAIN Function datafilled as a TOPS automated system.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

OAIN303

## OAINNODE

### Description

OM group OSSAIN Node Maintenance (OAINNODE) pegs state changes for all Operator Services System Advanced Intelligent Network (OSSAIN) nodes including Operator Services Node-Maintenance (OSNM), Operator Services Node (OSN), and Operator Services Systems Advanced Intelligent Network Centralization (OSAC) nodes.

The following table lists the key and info fields associated with OM group OAINNODE:

| Key field                                                                                                                            | Info field                                                         |
|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| OAINNODE can be indexed by either of the following: NODEID {0 to 767}: Key field for OANODINV.NODENAME: Name associated with NODEID. | OAINNODE_INDEX_OMINFO - This name can be up to 16 characters long. |

### Related functional groups

Enhanced Services (ENSV0001) is associated with OM group OAINNODE.

### Registers

The following table lists the registers associated with OM group OAINNODE and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAINNODE (Sheet 1 of 2)

| Register name            | Measures                |
|--------------------------|-------------------------|
| <a href="#">NAUDFAIL</a> | Node Audit Fail         |
| <a href="#">NDINSV</a>   | Node in Service         |
| <a href="#">NDISTB</a>   | Node in Service Trouble |
| <a href="#">NDMANB</a>   | Node Manual Busy        |
| <a href="#">NDSYSB</a>   | Node System Busy        |



**Registers for OM group OAINNODE (Sheet 2 of 2)**

| Register name            | Measures       |
|--------------------------|----------------|
| <a href="#">NRTSFAIL</a> | Node RTS Fail  |
| <a href="#">NTSTFAIL</a> | Node Test Fail |

**NAUDFAIL****Register type**

Peg

**Description**

NAUDFAIL is pegged when the node goes system busy due to audit failure.

**Associated registers**[NDSYSB](#)**Extension registers**

None

**Associated logs**

PM102

**NDINSV****Register type**

Peg

**Description**

NDINSV is pegged when the node is brought into service.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM106

**NDISTB****Register type**

Peg

**Description**

NDISTB is pegged when the node goes ISTB due to the session pool going out of service.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM128

**NDMANB****Register type**

Peg

**Description**

NDMANB is pegged when the node is manually busied from the MAP.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM105

**NDSYSB****Register type**

Peg

**Description**

NDSYSB is pegged under the following conditions:

- The node goes system busy due to audit failure.
- The node goes system busy due to RTS failure.
- The node goes system busy due to a request from the remote node.

**Associated registers**

[NAUDFAIL](#), [NRTSFAIL](#)

**Extension registers**

None

**Associated logs**

PM102

**NRTSFAIL****Register type**

Peg

**Description**

NRTSFAIL is pegged when the node goes system busy due to RTS failure.

**Associated registers**[NDSYSB](#)**Extension registers**

None

**Associated logs**

PM102

**NTSTFAIL****Register type**

Peg

**Description**

NTSTFAIL is pegged when the node fails a manual test.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM100

## OAINQMS

### Description

OM group Operator Services System Advanced Intelligent Network (OSSAIN) Queue Management System (OAINQMS) provides peg counts for OSSAIN calls on a per queue basis. It provides counts for calls that request an OSSAIN session from the QMS call agent and manager (CAM) and also counts on the action taken by the CAM in response to the request.

The following table lists the key and info fields associated with OM group OAINQMS:

| Key field                       | Info field |
|---------------------------------|------------|
| OSSAIN Call Queue (CQ0 - CQ254) | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAINQMS.

### Registers

The following table lists the registers associated with OM group OAINQMS and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAINQMS (Sheet 1 of 2)

| Register name            | Measures                                 |
|--------------------------|------------------------------------------|
| <a href="#">ABANDONC</a> | Call Abandoned while in queue            |
| <a href="#">DEFLCTCQ</a> | Call Deflected                           |
| <a href="#">DENIEDCQ</a> | Call Queuing Denied                      |
| <a href="#">GOTSESIM</a> | Got Session Immediately                  |
| <a href="#">OVFLMXAP</a> | Call Overflowed (No Call Queue Elements) |
| <a href="#">OVFLMXCQ</a> | Call Overflowed (MAXSIZE exceeded)       |

**Registers for OM group OAINQMS (Sheet 2 of 2)**

| Register name            | Measures                          |
|--------------------------|-----------------------------------|
| <a href="#">QUEUEDC</a>  | Call Queued                       |
| <a href="#">SESRQSTD</a> | Session Requested for OSSAIN call |

**ABANDONC****Register type**

Peg

**Description**

Pegged when an OSSAIN call is abandoned by a subscriber while the call is in queue for a session.

**Note:** For test case(s), make an OSSAIN call when no sessions to a service node are available and have the calling line go on hook while the call is queued.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**DEFLCTCQ****Register type**

Peg

**Description**

Pegged when a call destined for a call queue is deflected by the QMS CAM because no agent is available to serve the call, and the projected wait time for the call exceeds CDTIME datafilled for the queue in table QMSCQDEF.

**Note:** For test case(s), make an OSSAIN call when no sessions to a service node are available, and ensure that the predicted wait time for the queue exceeds the CDTIME value datafilled in table QMSCQDEF for the queue.

**Associated registers**

[SESRQSTD](#), [QUEUEDC](#), [GOTSESIM](#), [OVFLMXCQ](#), [OVFLMXAP](#),  
[DENIEDCQ](#)

**Validation formula**

$\text{SESRQSTD} = \text{QUEUEDC} + \text{GOTSESIM} + \text{DEFLCTCQ} + \text{OVFLMXCQ} + \text{OVFLMXAP} + \text{DENIEDCQ}$

**Extension registers**

None

**Associated logs**

None

**DENIEDCQ****Register type**

Peg

**Description**

Pegged when a call destined for a call queue is deflected by the QMS CAM because no agent is available to serve the call, and the QMS CAM is unable to queue the call for reasons other than those described for registers DEFLCTCQ, OVFLMXCQ, and OVFLMXAP.

**Note:** There are no test case(s) for this register.

**Associated registers**

[SESRQSTD](#), [QUEUEDC](#), [GOTSESIM](#), [OVFLMXCQ](#), [OVFLMXAP](#),  
[DEFLCTCQ](#)

**Validation formula**

$\text{SESRQSTD} = \text{QUEUEDC} + \text{GOTSESIM} + \text{DEFLCTCQ} + \text{OVFLMXCQ} + \text{OVFLMXAP} + \text{DENIEDCQ}$

**Extension registers**

None

**Associated logs**

None

**GOTSESIM****Register type**

Peg

**Description**

Pegged when a session is obtained immediately from the QMS CAM on request.

**Note:** For test case(s), make an OSSAIN call and ensure that a session is immediately obtained.

**Associated registers**

[SESRQSTD](#), [QUEUEDC](#), [DENIEDCQ](#), [OVFLMXCQ](#), [OVFLMXAP](#), [DEFLCTCQ](#)

**Validation formula**

$$\text{SESRQSTD} = \text{QUEUEDC} + \text{GOTSESIM} + \text{DEFLCTCQ} + \text{OVFLMXCQ} + \text{OVFLMXAP} + \text{DENIEDCQ}$$

**Extension registers**

GOTSESI2

**Associated logs**

None

**OVFLMXAP****Register type**

Usage

**Description**

Pegged when a call destined for a call queue is overflowed by the QMS CAM because no agent is available to serve the call, and the call queuing elements for the application have been exhausted (as specified by datafill in table QAPLNDEF).

**Note:** For test case(s), make an OSSAIN call when no sessions to a service node are available. Ensure that the number of calls in queue for the OSSAIN application equals the value datafilled by field CQELEMS, in table QAPLNDEF.

**Associated registers**

[SESRQSTD](#), [QUEUEDC](#), [GOTSESIM](#), [OVFLMXCQ](#), [DENIEDCQ](#), [DEFLCTCQ](#)

**Validation formula**

$$\text{SESRQSTD} = \text{QUEUEDC} + \text{GOTSESIM} + \text{DEFLCTCQ} + \text{OVFLMXCQ} + \text{OVFLMXAP} + \text{DENIEDCQ}$$

**Extension registers**

None

**Associated logs**

None

**OVFLMXCQ****Register type**

Peg

**Description**

Pegged when a call destined for a call queue is overflowed by the QMS CAM because no agent is available to serve the call, and the number of calls in the call queue that the call was destined for exceeds the MAXSIZE value datafilled for the queue in table QMSCQDEF.

**Note:** For test case(s), make an OSSAIN call when no sessions to a service node are available. Ensure that the number of calls in queue for the OSSAIN application equals the value datafilled by field CQELEMS, in table QAPLNDEF.

**Associated registers**

[SESRQSTD](#), [QUEUEDC](#), [GOTSESIM](#), [DENIEDCQ](#), [OVFLMXAP](#), [DEFLCTCQ](#)

**Validation formula**

$SESRQSTD = QUEUEDC + GOTSESIM + DEFLCTCQ + OVFLMXCQ + OVFLMXAP + DENIEDQ$

**Extension registers**

None

**Associated logs**

None

**QUEUEDC****Register type**

Peg

**Description**

Pegged when an OSSAIN call is queued for a session by the QMS CAM.

**Note:** For test case(s), make an OSSAIN call that must be queued for connectivity to a service node.

**Associated registers**

[SESRQSTD](#), [DENIEDCQ](#), [GOTSESIM](#), [OVFLMXCQ](#), [OVFLMXAP](#), [DEFLCTCQ](#)



**Validation formula**

SESRQSTD = QUEUEDC + GOTSESIM + DEFLCTCQ + OVFLMXCQ  
+ OVFLMXAP + DENIEDCQ

**Extension registers**

QUEUEDC2

**Associated logs**

None

**SESRQSTD****Register type**

Peg

**Description**

Pegged when a session is requested by an OSSAIN call from the QMS CAM.

**Note:** For test case(s), make an OSSAIN call that requires connectivity to a service node.

**Associated registers**

[DENIEDCQ](#), [QUEUEDC](#), [GOTSESIM](#), [OVFLMXCQ](#), [OVFLMXAP](#),  
[DEFLCTCQ](#)

**Validation formula**

SESRQSTD = QUEUEDC + GOTSESIM + DEFLCTCQ + OVFLMXCQ  
+ OVFLMXAP + DENIEDCQ

**Extension registers**

SESRQST2

**Associated logs**

None

## OAINRTE

### Description

OM group OSSAIN Route (OAINRTE) provides peg counts for obtaining sessions from session pools used for host-remote sessions or trigger event informs.

The following table lists the key and info fields associated with OM group OAINRTE:

| Key field                                                                         | Info field                                                                  |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| OAINRTE can be indexed by the following: SESNPLNM: Name associated with SESNPLID. | OAINRTE_INDEX_REGISTERI<br>NFO - This name can be up to 16 characters long. |

### Related functional groups

Enhanced Services (ENSV0001) is associated with OM group OAINRTE.

### Registers

The following table lists the registers associated with OM group OAINRTE and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAINRTE

| Register name           | Measures                              |
|-------------------------|---------------------------------------|
| <a href="#">OSCGOTS</a> | OSAC Got Session                      |
| <a href="#">OSCOVFL</a> | OSAC Session Overflow                 |
| <a href="#">OSCSESQ</a> | OSAC Session Request                  |
| <a href="#">TRGGOTS</a> | Trigger Event Inform Got Session      |
| <a href="#">TRGOVFL</a> | Trigger Event Inform Session Overflow |
| <a href="#">TRGSESQ</a> | Trigger Event Inform Session Request  |

#### OSCGOTS

Register type  
Peg

**Description**

OSCGOTS is pegged each time a host-remote session is obtained.

**Associated registers**

[OSCSESQ](#), [OSCOVFL](#)

**Extension registers**

OSCGOTS2

**Associated logs**

None

**OSCOVFL****Register type**

Peg

**Description**

OSCOVFL is pegged each time a host-remote session is requested but there are no sessions available.

**Associated registers**

[OSCSESQ](#), [OSCGOTS](#)

**Extension registers**

None

**Associated logs**

None

**OSCSESQ****Register type**

Peg

**Description**

OSCSESQ is pegged each time a host-remote session is requested.

**Associated registers**

[OSCGOTS](#), [OSCOVFL](#)

**Extension registers**

OSCSESQ2

**Associated logs**

None

**TRGGOTS****Register type**

Peg

**Description**

TRGGOTS is pegged each time a session for a trigger event inform is obtained.

**Associated registers**

[TRGSESQ](#), [TRGOVFL](#)

**Extension registers**

TRGGOTS2

**Associated logs**

None

**TRGOVFL****Register type**

Peg

**Description**

TRGOVFL is pegged each time a session for a trigger event is requested but there are no sessions available.

**Associated registers**

[TRGSESQ](#), [TRGGOTS](#)

**Extension registers**

None

**Associated logs**

None

**TRGSESQ****Register type**

Peg

**Description**

TRGSESQ is pegged each time a session is requested from a session pool used only for trigger event informs.

**Associated registers**

[TRGGOTS](#), [TRGOVFL](#)

**Extension registers**

TRGSESQ2

**Associated logs**

None

## OANODEDC

### Description

OM group Operator Services System Advanced Intelligent Network (OSSAIN) Node Data Communications (OANODEDC) is created for data communications operational measurements:

- OADATCOM - OSSAIN Data Communications
- OASNPLDC - OSSAIN Session Pool Data Communications

OM group OANODEDC provides peg counts for OSSAIN data communications messaging events on a per node basis. It provides counts for the total number of messages sent from the CM to a each external node and the total number of messages received by the CM from each external node. Counts of messages are broken down into successful and failure counts.

The following table lists the key and info fields associated with OM group OANODEDC:

| Key field                                      | Info field                  |
|------------------------------------------------|-----------------------------|
| NODEID {0 - 31}: Key field from table OANODINV | OSSAIN_NODE_DATACOM_O MINFO |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OANODEDC.

### Registers

The following table lists the registers associated with OM group OANODEDC and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OANODEDC (Sheet 1 of 2)

| Register name            | Measures                                |
|--------------------------|-----------------------------------------|
| <a href="#">ONMSGRCV</a> | OSSAIN message received per node        |
| <a href="#">ONMSGRFL</a> | OSSAIN message receive failure per node |
| <a href="#">ONMSGRSC</a> | OSSAIN message receive success per node |

**Registers for OM group OANODEDC (Sheet 2 of 2)**

| Register name            | Measures                                      |
|--------------------------|-----------------------------------------------|
| <a href="#">ONMSGSFL</a> | OSSAIN message send failure per node          |
| <a href="#">ONMSGSND</a> | OSSAIN message send requested per node        |
| <a href="#">ONMSGSSC</a> | OSSAIN message send success per node          |
| <a href="#">ONRCRTFL</a> | OSSAIN message receive route failure per node |
| <a href="#">ONSNRTFL</a> | OSSAIN message receive route failure per node |

**ONMSGRCV****Register type**

Peg

**Description**

ONMSGRCV is pegged for a specific node each time an incoming message, originating from an external node, is received from that node. This includes both call processing and maintenance messages.

**Associated registers**[ONMSGRSC](#), [ONMSGRFL](#), OSMSGRC**Validation formula**
$$\text{ONMSGRCV} = \text{ONMSGRSC} + \text{ONMSGRFL}$$
**Extension registers**

ONMSGRC2

**Associated logs**

None

**ONMSGRFL****Register type**

Peg

**Description**

ONMSGRFL is pegged for a specific node each time data communications encounters an error while attempting to forward a message originated from that node to the destination DMS process. This can be caused by a failure in the DMS internal messaging system or data transport interface. This register is also pegged during failures indicated by register ONRCRTFL.

**Associated registers**

[ONMSGRCV](#), [ONRCRTFL](#), [ONMSGRSC](#), OSMSGRFL

**Validation formulas**

The following formulas relate register ONMSGRFL and its associated registers:

- $\text{ONMSGRCV} = \text{ONMSGRSC} + \text{ONMSGRFL}$
- $\text{ONMSGRFL} \geq \text{ONRCRTFL}$

**Extension registers**

None

**Associated logs**

OAIN605, OAIN606

**ONMSGRSC****Register type**

Peg

**Description**

ONMSGRSC is pegged for a specific node when the CM's data communications software is able to successfully process an incoming message from the node.

**Associated registers**

[ONMSGRCV](#), [ONMSGRFL](#), OSMSGRSC

**Validation formula**

$\text{ONMSGRSC} = \text{ONMSGRCV} - \text{ONMSGRFL}$

**Extension registers**

ONMSGRS2

**Associated logs**

None

**ONMSGSFL****Register type**

Peg

**Description**

ONMSGSFL is pegged for a specific node each time data communications encounters an error while attempting to send an outgoing message to the node. This can be caused by a transport layer failure. This register is also pegged for reasons indicated by register ONSNRTFL.



**Associated registers**

[ONMSG SND](#), [ONMSG SSC](#), [ONSNRTFL](#), OSMSG SFL

**Validation formulas**

The following formulas relate register ONMSG SFL and its associated registers:

- $ONMSG SFL = ONMSG SND + ONMSG SSC$
- $ONMSG SFL \geq ONSNRTFL$

**Extension registers**

None

**Associated logs**

OAIN607, OAIN706

**ONMSG SND****Register type**

Peg

**Description**

ONMSG SND is pegged for a specific node each time the data communications software is requested to send a message. This includes requests from call processes and maintenance processes.

**Associated registers**

[ONMSG SSC](#), [ONMSG SFL](#), OSMSG SN

**Validation formula**

$ONMSG SND = ONMSG SSC + ONMSG SFL$

**Extension registers**

ONMSG SN2

**Associated logs**

None

**ONMSG SSC****Register type**

Peg

**Description**

ONMSG SSC is pegged for a specific node when the CM's data communications software is able to successfully process an outgoing message destined for that node. Note that OSSAIN uses unguaranteed messaging. Pegging this register does not indicate that the message actually arrived at the destination node.

**Associated registers**

[ONMSGSSND](#), [ONMSGSF](#), OSMSGSSC

**Validation formula**

ONMSGSSC = ONMSGSSND - ONMSGSF

**Extension registers**

ONMSGSS2

**Associated logs**

None

**ONRCRTFL****Register type**

Peg

**Description**

ONRCRTFL is pegged for a specific node each time the data communications software is unable to determine the destination of a message originating from that node. This can be caused by a variety of reasons including:

- invalid protocol version
- invalid session pool identifier
- invalid session identifier
- invalid network address
- invalid session pool state
- invalid node state
- pool / node identifier mis-match
- corrupted message

**Associated registers**

[ONMSGF](#), OSRCRTFL

**Validation formula**

ONRCRTFL <= ONMSGF

**Extension registers**

None

**Associated logs**

OAIN605, OAIN606

**ONSNRTFL****Register type**

Peg

**Description**

ONSNRTFL is pegged for a specific node each time the data communications software is unable to determine the destination of an outgoing message. This can be caused by the following reasons:

- invalid session pool identifier
- invalid session identifier
- pool/node identifier mis-match
- corrupted data

**Associated registers**[ONMSGSF](#), ONSNRTFL**Validation formula**
$$\text{ONSNRTFL} < \text{ONMSGSF}$$
**Extension registers**

None

**Associated logs**

None

## OAPCALP1

### Description

OM group Open Automated Protocol (OAP) Call Processing 1 (OAPCALP1) contains a register for each call processing and non-call processing operation and response message defined in the OAP protocol. The purpose of the registers in this OM group is to track usage of the operations and responses. These OM registers are pegged on a per session pool basis for call processing and session pool operations and are pegged on a per node basis for node maintenance operations.

The following table lists the key and info fields associated with OM group OAPCALP1:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP1.

### Registers

The following table lists the registers associated with OM group OAPCALP1 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPCALP1 (Sheet 1 of 2)

| Register name           | Measures                           |
|-------------------------|------------------------------------|
| <a href="#">APDAMAE</a> | Append AMA Module Error Response   |
| <a href="#">APDAMAS</a> | Append AMA Module Success Response |
| <a href="#">APNDAMA</a> | Append AMA Module Request          |
| <a href="#">BILNUME</a> | Billing Number Error Response      |
| <a href="#">BILNUMS</a> | Billing Number Success Response    |
| <a href="#">BLNGNUM</a> | Billing Number Request             |
| <a href="#">CALDETE</a> | Call Details Error Response        |

**Registers for OM group OAPCALP1 (Sheet 2 of 2)**

| Register name           | Measures                             |
|-------------------------|--------------------------------------|
| <a href="#">CALDETS</a> | Call Details Success Response        |
| <a href="#">CALLDET</a> | Call Details Request                 |
| <a href="#">CALLEND</a> | Call End Inform                      |
| <a href="#">CLSCHGE</a> | Class Change Error Response          |
| <a href="#">CLSCHGS</a> | Class Charge Success Response        |
| <a href="#">CLSCHRG</a> | Class Charge Request                 |
| <a href="#">GENAMA</a>  | Generate AMA Record                  |
| <a href="#">GENAMAE</a> | Generate AMA Record Error Response   |
| <a href="#">GENAMAS</a> | Generate AMA Record Success Response |

**APDAMAE****Register type**

Peg

**Description**

APDAMAE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

APDAMAE2

**Associated logs**

None

**APDAMAS****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

APDAMAS2

**Associated logs**

None

**APNDAMA****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

APNDAMA2

**Associated logs**

None

**BILNUME****Register type**

Peg

**Description**

BILNUME is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

BILNUME2

**Associated logs**

None

**BILNUMS****Register type**

Peg

**Description**

BILNUMS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

BILNUMS2

**Associated logs**

None

**BLNGNUM****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

BLNGNUM2

**Associated logs**

None

**CALDETE****Register type**

Peg

**Description**

CALDETE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALDETE2

**Associated logs**

None

**CALDETS****Register type**

Peg

**Description**

CALDETS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALDETS2

**Associated logs**

None

**CALLDET****Register type**

Peg



**Description**

CALLDET is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALLDET2

**Associated logs**

None

**CALLEND****Register type**

Peg

**Description**

CALLEND is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALLEND2

**Associated logs**

None

**CLSCHGE****Register type**

Peg

**Description**

CLSCHGE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CLSCHGE2

**Associated logs**

None

**CLSCHGS****Register type**

Peg

**Description**

CLSCHGS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CLSCHGS2

**Associated logs**

None

**CLSCHR****Register type**

Peg

**Description**

CLSCHR is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CLSCHR2

**Associated logs**

None

**GENAMA****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

GENAMA2

**Associated logs**

None

**GENAMAE****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

GENAMAE2

**Associated logs**

None

**GENAMAS****Register type**

Peg

**Description**

This register is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

GENAMAS2

**Associated logs**

None

## OAPCALP2

### Description

OM group Open Automated Protocol (OAP) Call Processing 2 (OAPCALP2) contains a register for each call processing and non-call processing operation and response message defined in the OAP protocol. The purpose of the registers in this OM group is to track usage of the operations and responses. These OM registers are pegged on a per session pool basis for call processing and session pool operations and are pegged on a per node basis for node maintenance operations.

The following table lists the key and info fields associated with OM group OAPCALP2:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP2.

### Registers

The following table lists the registers associated with OM group OAPCALP2 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPCALP2 (Sheet 1 of 2)

| Register name           | Measures                    |
|-------------------------|-----------------------------|
| <a href="#">CALFLTE</a> | Call Float Error Response   |
| <a href="#">CALFLTS</a> | Call Float Success Response |
| <a href="#">CALLFLT</a> | Call Float Request          |
| <a href="#">CONDN</a>   | Connect DN Request          |
| <a href="#">CONDNE</a>  | Connect DN Error Response   |
| <a href="#">CONDNS</a>  | Connect DN Success Response |
| <a href="#">CONSTAT</a> | Connection Status Inform    |

**Registers for OM group OAPCALP2 (Sheet 2 of 2)**

| Register name           | Measures                          |
|-------------------------|-----------------------------------|
| <a href="#">DIRNUM</a>  | Directory Number Request          |
| <a href="#">DIRNUME</a> | Directory Number Error Response   |
| <a href="#">DIRNUMS</a> | Directory Number Success Response |
| <a href="#">ENDCALE</a> | End Call Error Response           |
| <a href="#">ENDCALL</a> | End Call Request                  |
| <a href="#">ENDCALS</a> | End Call Success Response         |
| <a href="#">RELSDN</a>  | Release DN Request                |
| <a href="#">RELSDNE</a> | Release DN Error Response         |
| <a href="#">RELSDNS</a> | Release DN Success Response       |

**CALFLTE****Register type**

Peg

**Description**

CALFLTE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALFLTE2

**Associated logs**

None

**CALFLTS****Register type**

Peg

**Description**

CALFLTS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALFLTS2

**Associated logs**

None

**CALLFLT****Register type**

Peg

**Description**

CALLFLT is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CALLFLT2

**Associated logs**

None

**CONDN****Register type**

Peg

**Description**

CONDN is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CONDN2

**Associated logs**

None

**CONDNE****Register type**

Peg

**Description**

CONDNE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CONDNE2

**Associated logs**

None

**CONDNS****Register type**

Peg

**Description**

CONDNS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CONDNS2



**Associated logs**

None

**CONSTAT****Register type**

Peg

**Description**

CONSTAT is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CONSTAT2

**Associated logs**

None

**DIRNUM****Register type**

Peg

**Description**

CALDETS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

DIRNUM2

**Associated logs**

None

**DIRNUME****Register type**

Peg

**Description**

DIRNUME is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

DIRNUME2

**Associated logs**

None

**DIRNUMS****Register type**

Peg

**Description**

DIRNUMS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

DIRNUMS2

**Associated logs**

None

**ENDCALE****Register type**

Peg

**Description**

ENDCALE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

ENDCALE2

**Associated logs**

None

**ENDCALL****Register type**

Peg

**Description**

ENDCALL is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

ENDCALL2

**Associated logs**

None

**ENDCAL****Register type**

Peg

**Description**

ENDCAL is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

ENDCAL2

**Associated logs**

None

**RELSDN****Register type**

Peg

**Description**

RELSDN is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELSDN2

**Associated logs**

None

**RELSDNE****Register type**

Peg

**Description**

RELSDNE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELSDNE2

**Associated logs**

None

**RELSDNS****Register type**

Peg

**Description**

RELSDNS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELSDNS2

**Associated logs**

None

## OAPCALP3

### Description

OM group Open Automated Protocol (OAP) Call Processing 3 (OAPCALP3) contains a register for each call processing and non-call processing operation and response message defined in the OAP protocol. The purpose of the registers in this OM group is to track usage of the operations and responses. These OM registers are pegged on a per session pool basis for call processing and session pool operations and are pegged on a per node basis for node maintenance operations.

The following table lists the key and info fields associated with OM group OAPCALP3:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP3.

### Registers

The following table lists the registers associated with OM group OAPCALP3 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPCALP3 (Sheet 1 of 2)

| Register name           | Measures                            |
|-------------------------|-------------------------------------|
| <a href="#">SESNBEG</a> | Session Begin Inform                |
| <a href="#">SESNINE</a> | Session Initiation Error Response   |
| <a href="#">SESNINI</a> | Session Initiation Request          |
| <a href="#">SESNINS</a> | Session Initiation Success Response |
| <a href="#">SPCHPTE</a> | Speech Path Error Response          |
| <a href="#">SPCHPTH</a> | Speech Path Request                 |
| <a href="#">SPCHPTS</a> | Speech Path Success Response        |

**Registers for OM group OAPCALP3 (Sheet 2 of 2)**

| Register name           | Measures                                  |
|-------------------------|-------------------------------------------|
| <a href="#">TRGEVT</a>  | Trigger Event Inform                      |
| <a href="#">TXTOPR</a>  | Text to Operator Request                  |
| <a href="#">TXTOPRE</a> | Text to Operator Error Response           |
| <a href="#">TXTOPRS</a> | Text to Operator Success Response         |
| <a href="#">XFRCTRE</a> | Transfer to Control List Error Response   |
| <a href="#">XFRCTRL</a> | Transfer to Control List Request          |
| <a href="#">XFRCTRS</a> | Transfer to Control List Success Response |

**SESNBEG****Register type**

Peg

**Description**

SESNBEG is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SESNBEG2

**Associated logs**

None

**SESNINE****Register type**

Peg

**Description**

SESNINE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SESNINE2

**Associated logs**

None

**SESNINI****Register type**

Peg

**Description**

SESNINI is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SESNINI2

**Associated logs**

None

**SESNINS****Register type**

Peg

**Description**

SESNINS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.



**Associated registers**

None

**Extension registers**

SESNINS2

**Associated logs**

None

**SPCHPTE****Register type**

Peg

**Description**

SPCHPTE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SPCHPTE2

**Associated logs**

None

**SPCHPTH****Register type**

Peg

**Description**

SPCHPTH is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SPCHPTH2

**Associated logs**

None

**SPCHPTS****Register type**

Peg

**Description**

SPCHPTS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

SPCHPTS2

**Associated logs**

None

**TRGEVT****Register type**

Peg

**Description**

TRGEVT is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

TRGEVT2

**Associated logs**

None

**TXTOPR****Register type**

Peg

**Description**

TXTOPR is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

TXTOPR2

**Associated logs**

None

**TXTOPRE****Register type**

Peg

**Description**

TXTOPRE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

TXTOPRE2

**Associated logs**

None

**TXTOPRS****Register type**

Peg

**Description**

TXTOPRS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

TXTOPRS2

**Associated logs**

None

**XFRCTRE****Register type**

Peg

**Description**

XFRCTRE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

XFRCTRE2

**Associated logs**

OAIN203

**XFRCTRL****Register type**

Peg

**Description**

XFRCTRL is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

XFRCTRL2

**Associated logs**

None

**XFRCTRS****Register type**

Peg

**Description**

XFRCTRS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

XFRCTRS2

**Associated logs**

None

## OAPCALP4

### Description

OM group Open Automated Protocol (OAP) Call Processing 4 (OAPCALP4) contains a register for each call processing and non-call processing operation and response message defined in the OAP protocol. The purpose of the registers in this OM group is to track usage of the operations and responses. These OM registers are pegged on a per session pool basis for call processing and session pool operations and are pegged on a per node basis for node maintenance operations.

The following table lists the key and info fields associated with OM group OAPCALP4:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP4.

### Registers

The following table lists the registers associated with OM group OAPCALP4 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPCALP4 (Sheet 1 of 2)

| Register name           | Measures                            |
|-------------------------|-------------------------------------|
| <a href="#">CARASGN</a> | Carrier Assignment Request          |
| <a href="#">CARASNE</a> | Carrier Assignment Error Response   |
| <a href="#">CARASNS</a> | Carrier Assignment Success Response |
| <a href="#">DTMFDIG</a> | DTMF Digit Detected Inform          |
| <a href="#">RELRCVE</a> | Release Receiver Error Response     |
| <a href="#">RELRCVR</a> | Release Receiver Request            |
| <a href="#">RELRCVS</a> | Release Receiver Success Response   |

**Registers for OM group OAPCALP4 (Sheet 2 of 2)**

| Register name           | Measures                            |
|-------------------------|-------------------------------------|
| <a href="#">RTETRMT</a> | Route To Treatment Request          |
| <a href="#">RTETRME</a> | Route To Treatment Error Response   |
| <a href="#">RTETRMS</a> | Route To Treatment Success Response |
| <a href="#">VCECON</a>  | Voice Connect Request               |
| <a href="#">VCECONE</a> | Voice Connect Error Response        |
| <a href="#">VCECONS</a> | Voice Connect Success Response      |
| <a href="#">VCERLS</a>  | Voice Release Request               |
| <a href="#">VCERLSE</a> | Voice Release Error Response        |
| <a href="#">VCERLSS</a> | Voice Release Success Response      |

**CARASGN****Register type**

Peg

**Description**

CARASGN is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CARASGN2

**Associated logs**

None

**CARASNE****Register type**

Peg

**Description**

CARASNE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CARASNE2

**Associated logs**

None

**CARASNS****Register type**

Peg

**Description**

CARASNS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

CARASNS2

**Associated logs**

None

**DTMFDIG****Register type**

Peg

**Description**

DTMFDIG is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.



**Associated registers**

None

**Extension registers**

DTMFDIG2

**Associated logs**

None

**RELRCVE****Register type**

Peg

**Description**

RELRCVE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELRCVE2

**Associated logs**

None

**RELRCVR****Register type**

Peg

**Description**

RELRCVR is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELRCVR2

**Associated logs**

None

**RELRCVS****Register type**

Peg

**Description**

RELRCVS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RELRCVS2

**Associated logs**

None

**RTETRMT****Register type**

Peg

**Description**

RTETRMT is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RTETRMT2

**Associated logs**

None

**RTETRME****Register type**

Peg

**Description**

RTETRME is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RTETRME2

**Associated logs**

None

**RTETRMS****Register type**

Peg

**Description**

RTETRMS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

RTETRMS2

**Associated logs**

None

**VCECON****Register type**

Peg

**Description**

VCECON is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCECON2

**Associated logs**

None

**VCECONE****Register type**

Peg

**Description**

VCECONE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCECONE2

**Associated logs**

None

**VCECONS****Register type**

Peg

**Description**

VCECONS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCECONS2

**Associated logs**

None

**VCERLS****Register type**

Peg

**Description**

VCERLS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCERLS2

**Associated logs**

None

**VCERLSE****Register type**

Peg

**Description**

VCERLSE is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCERLSE2

**Associated logs**

None

**VCERLSS****Register type**

Peg

**Description**

VCERLSS is pegged each time the corresponding call processing operation or response is sent or received by the switch.

**Note:** For test case(s), make a call that would require the corresponding call processing operation or response.

**Associated registers**

None

**Extension registers**

VCERLSS2

**Associated logs**

None

---

## OAPCALP5

---

### Description

In TOPS06, OM group Open Automated Protocol (OAP) Call Processing 5 (OAPCALP5) is added to the set of OAP message OMs. This OM group contains registers that track call processing operations and responses used with the Operator Services Systems Advanced Intelligent Network (OSSAIN) Enhancements feature.

The OSSAIN Enhancements feature provides the following capabilities:

- OSSAIN transition to Traffic Operator Position System Multipurpose (TOPS MP) position - provides Text to Operator information to a TOPS MP operator.
- OSSAIN Equal Access (EA) Enhancements - provide additional EA handling capability when the call is at an OSSAIN service node (SN).
- Open Automated Protocol Enhancements
  - pass additional carrier information to the SN
  - provide support for transfer to the interLATA carrier functionality
- Custom Automatic Message Accounting (AMA) Enhancements - allow custom AMA modules to be appended to the AMA record for the TOPS Charge Adjust service.
- AABS Replacement - allows the automation of 0+ 3rd, collect, and credit card calls on the OSSAIN SN platform utilizing OAP.

**Note:** For more information about the OSSAIN Enhancements feature, please refer to the "OSSAIN" section of the Translations Guide.

In TOPS07, the following features add registers to OM group OAPCALP5:

- TOPS Local Number Portability (LNP) Call Processing - adds registers LNPREQ, LNPREQE, and LNPREQS with their respective extension registers. These registers track OAP Local Number Portability (LNP) request and response messages. For more information about the TOPS LNP Call Processing feature, please refer to the "TOPS LNP" section of the Translations Guide.
- OSSAIN Enhancements II - adds registers CONVTM, CONVTMSE, CONVTME, RESUME, RESUMEE, and RESUMES with their respective extension registers. For more information about the OSSAIN Enhancements II feature, please refer to the "OSSAIN Enhancements" section of the Translations Guide.

Registers CONVTM, CONVTME, and CONVTVS track OAP request and response messages relative to conversation timing information for calls. Registers RESUME, RESUMES, and RESUMEE track OAP request and response messages relative to OSSAIN preprocessing.

- Branding for TOPS via SPID - adds registers SPDREQ, SPDREQS, and SPDREQE. These registers are pegged for Call Processing class message operations on a per session pool basis and adds the necessary registers for the SPID assignment request, success response, and error response messages.

The following table lists the key and info fields associated with OM group OAPCALP5:

| Key field                                          | Info field                                                                                                                                                    |
|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | OAP_SP_INDEX_REGISTERIN<br>FO - This field contains the name associated with the SESNPLID field in table OASESNPL. This name can be up to 16 characters long. |

**Note:** The DMS switch adds one tuple to this OM group for each SESNPLID datafilled in table OASESNPL.

## Related functional groups

The following functional groups are associated with OM group OAPCALP5.

- Enhanced Services (ENSV0001)
- Operator Services Equal Access (OSEA0001)

## Registers

The following table lists the registers associated with OM group OAPCALP5 and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OAPCALP5 (Sheet 1 of 2)

| Register name           | Measures                           |
|-------------------------|------------------------------------|
| <a href="#">CONVTM</a>  | Conversation Timing Request        |
| <a href="#">CONVTME</a> | Conversation Timing Error Response |



**Registers for OM group OAPCALP5 (Sheet 2 of 2)**

| Register name            | Measures                             |
|--------------------------|--------------------------------------|
| <a href="#">CONVTMS</a>  | Conversation Timing Success Response |
| <a href="#">LNPREQ</a>   | LNP Request                          |
| <a href="#">LNPREQE</a>  | LNP Request Error Response           |
| <a href="#">LNPREQS</a>  | LNP Request Success Response         |
| <a href="#">RESUME</a>   | Call Resume Request                  |
| <a href="#">RESUMEE</a>  | Call Resume Request Error Response   |
| <a href="#">RESUMES</a>  | Call Resume Request Response         |
| <a href="#">SPDREQ</a>   | SPID Assignment Request              |
| <a href="#">SPDREQE</a>  | SPID Assignment Error Response       |
| <a href="#">SPDREQS</a>  | SPID Assignment Success Response     |
| <a href="#">XFRCAR</a>   | Transfer to Carrier Request          |
| <a href="#">XFR CARE</a> | Transfer to Carrier Error Response   |
| <a href="#">XFR CARS</a> | Transfer to Carrier Success Response |

**CONVTM****Register type**

Peg

**Description**

The DMS switch pegs register CONVTM each time it receives a conversation timing request from an active SN. An active SN initiates this request when it wants the DMS switch to start or stop conversation timing for a call.

**Associated registers**

None

**Extension registers**

CONVTM2

**Associated logs**

None

**CONVTME****Register type**

Peg

**Description**

The DMS switch pegs register CONVTME each time it sends a conversation timing error response to an active SN. This call processing response informs the active SN that the DMS switch could not process the conversation timing request; nor could it start or stop conversation timing for a call.

**Associated registers**

None

**Extension registers**

CONVTME2

**Associated logs**

None

**CONVTMS****Register type**

Peg

**Description**

The DMS switch pegs register CONVTMS each time it sends a conversation timing success response to an active SN. This call processing response informs the active SN that the DMS switch successfully processed the conversation timing request and that call timing can be successfully started or stopped.

**Associated registers**

None

**Extension registers**

CONVTMS2

**Associated logs**

None

**LNPREQ****Register type**

Peg

**Description**

The DMS switch pegs register LNPREQ each time it receives a LNP request from an active SN. An active SN initiates this request when it needs portability information about a directory number (DN).

**Associated registers**

None

**Extension registers**

LNPREQ2

**Associated logs**

None

**LNPREQE****Register type**

Peg

**Description**

The DMS switch pegs register LNPREQE each time it sends an LNP request error response to an active SN. This response informs the active SN that the LNP request failed.

**Associated registers**

None

**Extension registers**

LNPREQE2

**Associated logs**

None

**LNPREQS****Register type**

Peg

**Description**

The DMS switch pegs register LNPREQS each time it sends an LNP request success response to an active SN. This call processing response informs the active SN that the LNP request was processed successfully. Portability information is returned to the active SN.

**Associated registers**

None

**Extension registers**

LNPREQS2

**Associated logs**

None

**RESUME****Register type**

Peg

**Description**

The DMS switch pegs register RESUME each time it receives a call resume request from an active SN. An active SN initiates this request when it wants to release itself and its resources from an OSSAIN preprocessing call session, without terminating the call such that the DMS switch resumes control of the call.

**Associated registers**

None

**Extension registers**

RESUME2

**Associated logs**

None

**RESUMEE****Register type**

Peg

**Description**

The DMS switch pegs register RESUMEE each time it sends a call resume error response to an active SN. This call processing response informs the active SN that the DMS switch could not process the call resume request; nor could it resume control of the OSSAIN call.

**Associated registers**

None

**Extension registers**

RESUMEE2

**Associated logs**

None

**RESUMES****Register type**

Peg

**Description**

The DMS switch pegs register RESUMES each time it sends a call resume success response to an active SN. This call processing response informs the active SN that the DMS switch has successfully processed the call resume request and has successfully resumed control of the OSSAIN preprocessed call.

**Associated registers**

None

**Extension registers**

RESUMES2

**Associated logs**

None

**SPDREQ****Register type**

Peg

**Description**

SPDREQ is pegged each time the SPID Assignment Request operation is received by the switch.

**Associated registers**

None

**Extension registers**

SPDREQ2

**Associated logs**

None

**SPDREQE****Register type**

Peg

**Description**

SPDREQE is pegged each time the SPID Assignment Error Response is sent by the switch.

**Associated registers**

None

**Extension registers**

SPDREQE2

**Associated logs**

None

**SPDREQS****Register type**

Peg

**Description**

SPDREQS is pegged each time the SPID Assignment Success Response is sent by the switch.

**Associated registers**

None

**Extension registers**

SPDREQS2

**Associated logs**

None

**XFRCAR****Register type**

Peg

**Description**

The DMS switch pegs register XFRCAR each time it receives a transfer to carrier request from an active SN. An active SN initiates this request to request that the DMS switch release it from the call and transfer the call to the carrier.

**Associated registers**

None

**Extension registers**

XFRCAR2

**Associated logs**

None

**XFRCARE****Register type**

Peg

**Description**

The DMS switch pegs register XFRCARE each time it sends a transfer to carrier error response to an active SN. This call processing response

informs the active SN that the DMS switch could not process the transfer to carrier request; nor could it transfer the call to the carrier or release the SN from the call.

**Associated registers**

None

**Extension registers**

XFRCARE2

**Associated logs**

None

**XFRCARS****Register type**

Peg

**Description**

The DMS switch pegs register XFRCARS each time it sends a transfer to carrier success response to an active SN. This call processing response informs the active SN that the DMS switch has successfully processed the transfer to carrier request, released the active SN from the call, and has transferred the call to the carrier.

**Associated registers**

None

**Extension registers**

XFRCARS2

**Associated logs**

None

## OAPCALP6

### Description

In TOPS07, OM group Open Automated Protocol (OAP) Call Processing 6 (OAPCALP6) is added to the set of OAP message OMs. This OM group contains registers that track call processing operations and responses used with the Operator Services Systems Advanced Intelligent Network (OSSAIN) simultaneous interactions feature.

The OSSAIN simultaneous interactions feature allows the attachment of two OSSAIN function providers (service node or TOPS operator) to a call simultaneously. The attachment configurations are as follows:

- service node and service node
- service node and a TOPS operator

During simultaneous interactions of a call, only one function provider may control the call. This function provider is the active agent. The other function provider is the passive agent.

**Note 1:** In an OSSAIN simultaneous interaction, a service node must always be the active agent. An operator can never be the active agent when it is engaged in a simultaneous interaction with a service node.

**Note 2:** For more information about OAP, refer to the OSSAIN Open Automated Protocol Specification, NIS: Q235-1

The following table lists the key and info fields associated with OM group OAPCALP6:

| Key field                                          | Info field                                                                                                                            |
|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | OAP_SP_INDEX_REGISTERINFO - This field corresponds to the SESNPLNM field in table OASESNPL. The name can be up to 16 characters long. |

**Note:** The DMS switch adds one tuple to this OM group for each SESNPLID datafiled in table OASESNPL.

### Related functional groups

Enhanced Services (ENSV0001) is associated with OM group OAPCALP6.



## Registers

The following table lists the registers associated with OM group OAPCALP6 and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OAPCALP6

| Register name            | Measures                                   |
|--------------------------|--------------------------------------------|
| <a href="#">ACPCNTL</a>  | Accept Control Inform                      |
| <a href="#">NODEREL</a>  | Node Release Inform                        |
| <a href="#">PASCNTE</a>  | Pass Control Error Response                |
| <a href="#">PASCNTL</a>  | Pass Control Request                       |
| <a href="#">PASCNTS</a>  | Pass Control Success Response              |
| <a href="#">PASREQE</a>  | Passive Function Provider Error Response   |
| <a href="#">PASREQS</a>  | Passive Function Provider Success Response |
| <a href="#">PASSREQ</a>  | Passive Function Provider Request          |
| <a href="#">PASSTAT</a>  | Passive Node Status Inform                 |
| <a href="#">PASTHRU</a>  | Pass-Through Inform                        |
| <a href="#">RELNOD</a>   | Release Node Request                       |
| <a href="#">RELNODE</a>  | Release Node Request Error Response        |
| <a href="#">RELNODS</a>  | Release Node Success Response              |
| <a href="#">SESRECL</a>  | Session Recall Request                     |
| <a href="#">SESRECLE</a> | Session Recall Return Error                |
| <a href="#">SESRECLS</a> | Session Recall Return Result               |

#### ACPCNTL

##### Register type

Peg

##### Description

The DMS switch pegs register ACPCNTL each time it sends an accept control inform message to a passive agent. This call processing

message informs the passive agent that it has become the active service agent for a call.

**Associated registers**

None

**Extension registers**

ACPCNTL2

**Associated logs**

None

**NODEREL****Register type**

Peg

**Description**

The DMS switch pegs register NODEREL each time it sends a node release inform message to a function provider. This call processing message informs a function provider that it is released from the call. The DMS switch sends this message under the following circumstances:

- the active agent requests release of the passive agent

**Note:** If the passive agent is an operator, the DMS switch sends an OPP/ASCII call end message.

- an operator requests release of the active agent (by keying the release node function)

**Associated registers**

None

**Extension registers**

NODEREL2

**Associated logs**

None

**PASCNTE****Register type**

Peg

**Description**

The DMS switch pegs register PASCNTE each time it sends a pass control error response to the active agent that initiated the pass control

request. This call processing response informs the active agent that it cannot pass call control to the passive agent.

**Associated registers**

None

**Extension registers**

PASCNTE2

**Associated logs**

None

**PASCNTL****Register type**

Peg

**Description**

The DMS switch pegs register PASCNTL each time an active agent sends a pass control request to the DMS switch. This call processing request informs the DMS switch that the active agent wants to pass call control to the passive agent.

**Note:** An operator can never be the active agent while engaged in a simultaneous interaction; therefore, the pass control capability only applies when two service nodes are attached to a call simultaneously.

**Associated registers**

None

**Extension registers**

PASCNTL2

**Associated logs**

None

**PASCNTS****Register type**

Peg

**Description**

The DMS switch pegs register PASCNTS each time it sends a pass control success response to the active agent that initiated the pass control request. This call processing response indicates that the active agent has passed call control to the passive agent, thus swapping the roles of the function providers.

**Associated registers**

None

**Extension registers**

PASCNTS2

**Associated logs**

None

**PASREQE****Register type**

Peg

**Description**

The DMS switch pegs register PASREQE each time it sends a passive function provider error response to the active agent that initiated the passive function provider request. This call processing response informs the active agent that no passive function provider could be obtained for the call.

This call processing response indicates one of the following:

- A passive agent is not connected to the call.
- The call was not queued for connection to a passive agent.

**Associated registers**

None

**Extension registers**

PASREQE2

**Associated logs**

None

**PASREQS****Register type**

Peg

**Description**

The DMS switch pegs register PASREQS each time it sends a passive function provider success response to the active agent that initiated the passive function provider request. This call processing response indicates one of the following:

- A passive agent is connected to the call.
- The call is in queue, waiting for the attachment of a passive agent.

**Associated registers**

None

**Extension registers**

PASREQS2

**Associated logs**

None

**PASSREQ****Register type**

Peg

**Description**

The DMS switch pegs register PASSREQ each time it receives a passive function provider request from the active agent. The active agent uses this call processing message to request that a passive function provider be connected to the call.

**Associated registers**

None

**Extension registers**

PASSREQ2

**Associated logs**

None

**PASSTAT****Register type**

Peg

**Description**

The DMS switch pegs register PASSTAT each time it sends a passive node status inform message to the active agent. This call processing message informs the active agent of changes in the status of the passive agent.

**Associated registers**

None

**Extension registers**

PASSTAT2

**Associated logs**

None

**PASTHRU****Register type**

Peg

**Description**

The DMS switch pegs register PASTHRU each time it sends/receives a pass-through inform message to/from a function provider. This call processing message allows the active and passive agents to communicate. These message can be initiated/received by the active agent or the passive agent. The DMS switch routes these messages between the function providers.

**Associated registers**

None

**Extension registers**

PASTHRU2

**Associated logs**

None

**RELNOD****Register type**

Peg

**Description**

The DMS switch pegs register RELNOD each time it receives a release node request from a function provider. A function provider uses this call processing request to request either the release of itself or another function provider from a call.

**Associated registers**

None

**Extension registers**

RELNOD

**Associated logs**

None

**RELNODE****Register type**

Peg

**Description**

The DMS switch pegs register RELNODE each time it sends a release node error response to a function provider that either requested to release itself or another function provider from a call. This call processing response informs the requesting function provider that the DMS switch could not release the function provider specified in the release node request.

**Associated registers**

None

**Extension registers**

RELNODE2

**Associated logs**

None

**RELNODES****Register type**

Peg

**Description**

The DMS switch pegs register RELNODES each time it sends a release node success response to a function provider that is either requesting to release itself or another function provider from a call. This call processing response alerts the requesting function provider that the release was successful.

This call processing response indicates one of the following:

- The specified node was released.
- The call was taken out of the queue if it was queued for a passive agent, and the release node request specified that the passive node was to be released.

**Associated registers**

None

**Extension registers**

RELNODES2

**Associated logs**

None

**SESRECL****Register type**

Peg

**Description**

The DMS switch peps register SESRECL each time a session recall request is sent or received by the DMS switch. This exchange occurs between an OSSAIN node.

**Associated registers**

None

**Extension registers**

SESRECL2

**Associated logs**

None

**SESRECLE****Register type**

Peg

**Description**

The DMS switch peps register SESRECLE each time a session recall return error is sent or received by the DMS switch. This exchange occurs between the switch and an OSSAIN node.

**Associated registers**

None

**Extension registers**

SESRECLE2

**Associated logs**

None

**SESRECLS****Register type**

Peg

**Description**

The DMS switch peps register SESRECLS each time a session recall return result is sent or received by the DMS switch. This exchange occurs between the switch and an OSSAIN node.

**Associated registers**

None

**Extension registers**

SESRECLS2



**Associated logs**  
None

## OAPCALP7

### Description

In TOPS09, OM group Open Automated Protocol (OAP) Call Processing 7 (OAPCALP7) is added to the set of OAP message OMs. This OM group contains registers that track call processing operations and responses used with the Operator Services Systems Advanced Intelligent Network (OSSAIN) simultaneous interactions feature.

The OSSAIN simultaneous interactions feature allows the attachment of two OSSAIN function providers (service node or TOPS operator) to a call simultaneously. The attachment configurations are as follows:

- service node and service node
- service node and a TOPS operator

During simultaneous interactions of a call, only one function provider may control the call. This function provider is the active agent. The other function provider is the passive agent.

**Note 1:** In an OSSAIN simultaneous interaction, a service node must always be the active agent. An operator can never be the active agent when it is engaged in a simultaneous interaction with a service node.

**Note 2:** For more information about OAP, refer to the *OSSAIN Open Automated Protocol Specification*, NIS: Q235-1

The following table lists the key and info fields associated with OM group OAPCALP7:

| Key field                                          | Info field                                                                                                                            |
|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | OAP_SP_INDEX_REGISTERINFO - This field corresponds to the SESNPLNM field in table OASESNPL. The name can be up to 16 characters long. |

**Note:** The DMS switch adds one tuple to this OM group for each SESNPLID datafiled in table OASESNPL.

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP7.

## Registers

The following table lists the registers associated with OM group OAPCALP7 and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OAPCALP7

| Register name             | Measures                               |
|---------------------------|----------------------------------------|
| <a href="#">CNFADD</a>    | Conference Add Request                 |
| <a href="#">CNFADDE</a>   | Conference Add Return Error            |
| <a href="#">CNFADDS</a>   | Conference Add Return Result           |
| <a href="#">CNFCREMV</a>  | Conference Remove Request              |
| <a href="#">CNFCREMVE</a> | Conference Remove Request Error        |
| <a href="#">CNFCREMVS</a> | Conference Remove Request Result       |
| <a href="#">CNFCRET</a>   | Conference Create Request              |
| <a href="#">CNFCRETE</a>  | Conference Create Request Result Error |
| <a href="#">CNFCRETS</a>  | Conference Create Request Result       |
| <a href="#">CNFDELT</a>   | Conference Details Request             |
| <a href="#">CNFDELTE</a>  | Conference Details Return Error        |
| <a href="#">CNFDELTS</a>  | Conference Details Return Request      |
| <a href="#">CNFREL</a>    | Conference Release Request             |
| <a href="#">CNFRELE</a>   | Conference Release Return Error        |
| <a href="#">CNFRELS</a>   | Conference Release Return Result       |

#### CNFADD

##### Register type

Peg

##### Description

CNFADD is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFADD2

**Associated logs**

None

**CNFADDE****Register type**

Peg

**Description**

CNFADDE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFADDE2

**Associated logs**

None

**CNFADDS****Register type**

Peg

**Description**

CNFADDS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFADDS2

**Associated logs**

None

**CNFCREMV****Register type**

Peg

**Description**

CNFCREMV is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCREMV2

**Associated logs**

None

**CNFCREMVE****Register type**

Peg

**Description**

CNFCREMVE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCREMVE2

**Associated logs**

None

**CNFCREMVS****Register type**

Peg

**Description**

CNFCREMVS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCREMVS2

**Associated logs**

None

**CNFCRET****Register type**

Peg

**Description**

CNFCRET is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCRET2

**Associated logs**

None

**CNFCRETE****Register type**

Peg

**Description**

CNFCRETE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCRETE2

**Associated logs**

None

**CNFCRETS****Register type**

Peg

**Description**

CNFCRETS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFCRETS2

**Associated logs**

None

**CNFDELT****Register type**

Peg

**Description**

CNFDELT is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFDELT2

**Associated logs**

None

**CNFDELTE****Register type**

Peg

**Description**

CNFDELTE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFDELTE2

**Associated logs**

None

**CNFDELTS****Register type**

Peg

**Description**

CNFDELTS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFDELTS2

**Associated logs**

None

**CNFREL****Register type**

Peg

**Description**

CNFREL is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFREL2

**Associated logs**

None

**CNFRELE****Register type**

Peg

**Description**

CNFRELE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFRELE2

**Associated logs**

None



**CNFRELS****Register type**

Peg

**Description**

CNFRELS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CNFRELS2

**Associated logs**

None

## OAPCALP8

### Description

In TOPS09, OM group Open Automated Protocol (OAP) Call Processing 8 (OAPCALP8) is added to the set of OAP message OMs. This OM group contains registers that track call processing operations and responses used with the Operator Services Systems Advanced Intelligent Network (OSSAIN) simultaneous interactions feature.

The OSSAIN simultaneous interactions feature allows the attachment of two OSSAIN function providers (service node or TOPS operator) to a call simultaneously. The attachment configurations are as follows:

- service node and service node
- service node and a TOPS operator

During simultaneous interactions of a call, only one function provider may control the call. This function provider is the active agent. The other function provider is the passive agent.

**Note 1:** In an OSSAIN simultaneous interaction, a service node must always be the active agent. An operator can never be the active agent when it is engaged in a simultaneous interaction with a service node.

**Note 2:** For more information about OAP, refer to the *OSSAIN Open Automated Protocol Specification*, NIS: Q235-1

The following table lists the key and info fields associated with OM group OAPCALP8:

| Key field                                          | Info field                                                                                                                            |
|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | OAP_SP_INDEX_REGISTERINFO - This field corresponds to the SESNPLNM field in table OASESNPL. The name can be up to 16 characters long. |

### Related functional groups

ENSV Enhanced Services (ENSV0001) is associated with OM group OAPCALP8.

### Registers

The following table lists the registers associated with OM group

OAPCALP8 and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OAPCALP8

| Register name           | Measures                                                                                 |
|-------------------------|------------------------------------------------------------------------------------------|
| <a href="#">CGPNRQ</a>  | Integrated Services Digital Network User Part (ISUP) calling party number update request |
| <a href="#">CGPNRQE</a> | ISUP Calling Party Number Update Request Error Response                                  |
| <a href="#">CGPNRQS</a> | ISUP Calling Party Number Update Request Success Response                                |
| <a href="#">SVCCHG</a>  | Service Change Request                                                                   |
| <a href="#">SVCCHGE</a> | Service Change Request Error Response                                                    |
| <a href="#">SVCCHGS</a> | Service Change Request Success Response                                                  |
| <a href="#">VCERLSI</a> | Voice Release Inform                                                                     |

#### CGPNRQ

**Register type**

Peg

**Description**

CGPNRQ is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CGPNRQ2

**Associated logs**

None

#### CGPNRQE

**Register type**

Peg

**Description**

CGPNRQE is pegged each time the this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CGPNRQE2

**Associated logs**

None

**CGPNRQS****Register type**

Peg

**Description**

CGPNRQS is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

CGPNRQS2

**Associated logs**

None

**SVCCHG****Register type**

Peg

**Description**

SVCCHG is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

SVCCHG2

**Associated logs**

None

**SVCCHGE****Register type**

Peg

**Description**

SVCCHGE is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

SVCCHGE2

**Associated logs**

None

**SVCCHGS****Register type**

Peg

**Description**

SVCCHGS is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

SVCCHGS2

**Associated logs**

None

**VCERLSI****Register type**

Peg

**Description**

VCERLSI is pegged each time this operation or response is sent or received by the switch.

**Associated registers**

None

**Extension registers**

VCERLSI2

**Associated logs**

None

## OAPCALP9

### Description

OM group Open Automated Protocol (OAP) Call Processing 9 (OAPCALP9) contains registers that track call processing message operations and responses on an Operator Services Systems Advanced Intelligent Network (OSSAIN) session pool basis. A register is pegged each time the corresponding open automated protocol (OAP) message is sent or received by the DMS switch.

**Note:** For more information about OAP, refer to the *OSSAIN Open Automated Protocol Specification*, NIS: Q235-1

The following table lists the key and info fields associated with OM group OAPCALP9:

| Key field                                                                                                                                                                                                                                                                                          | Info field                                                            |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| The key field can be indexed by either of the following:<br>SESNPLID (0-4094) - This field corresponds to the key field SESNPLID in table OASESNPL.<br>SESNPLNM (up to 16 characters) - This field corresponds to field SESNPLNM in table OASESNPL. This field is a name associated with SESNPLID. | Call processing class message operations on a per session pool basis. |

**Note:** The DMS switch adds one tuple to this OM group for each SESNPLID datafilled in table OASESNPL.

### Related functional groups

The following functional groups are associated with OM group OAPCALP9:

- Enhanced Services, ENSV0001
- OSSAIN, OSAN0001

### Registers

The following table lists the registers associated with OM group OAPCALP9 and what they measure. For a detailed description of a

register, click on the register name.

### Registers for OM group OAPCALP9

| Register name           | Measures                         |
|-------------------------|----------------------------------|
| <a href="#">CNTTMT</a>  | Connect To Treatment Request     |
| <a href="#">CNTTMTE</a> | Connect To Treatment Error       |
| <a href="#">CNTTMTS</a> | Connect To Treatment Success     |
| <a href="#">ESTCHG</a>  | Estimate of Call Charges         |
| <a href="#">ESTCHGE</a> | Estimate of Call Charges Error   |
| <a href="#">ESTCHGS</a> | Estimate of Call Charges Success |
| <a href="#">PASTHRE</a> | Pass Through Error Response      |
| <a href="#">PASTHRQ</a> | Pass Through Request             |
| <a href="#">PASTHRS</a> | Pass Through Success Response    |
| <a href="#">SACTINE</a> | Service Active Inform            |

#### CNTTMT

**Register type**

Peg

**Description**

CNTTMT is pegged each time an OAP Connect To Treatment Request operation or response is sent or received by the switch.

To test this register, send or receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CNTTMT2

**Associated logs**

None



**CNTTMTE****Register type**

Peg

**Description**

CNTTMTE is pegged each time an OAP Connect To Treatment Error operation or response is sent or received by the switch.

To test this register, send or receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CNTTMTE2

**Associated logs**

None

**CNTTMTS****Register type**

Peg

**Description**

CNTTMTS is pegged each time an OAP Connect To Treatment Success operation or response is sent or received by the switch.

To test this register, send or receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CNTTMTS2

**Associated logs**

None

**ESTCHG****Register type**

Peg

**Description**

ESTCHG is pegged each time an estimate of charge operation or response is sent or received by the switch.

To test this register, send an estimate of charges to the switch and ensure that this register is pegged.

**Associated registers**

None

**Extension registers**

ESTCHG2

**Associated logs**

None

**ESTCHGE****Register type**

Peg

**Description**

ESTCHGE is pegged each time the estimate of charges error operation or response is sent or received by the switch.

To test this register, send an estimate of charges error operation to the switch and ensure that this register is pegged.

**Associated registers**

None

**Extension registers**

ESTCHGE2

**Associated logs**

None

**ESTCHGS****Register type**

Peg

**Description**

ESTCHGS is pegged each time the estimate of charges success operation or response is sent or received by the switch.

To test this register, send an estimate of charges success operation to the switch and ensure that this register is pegged.

**Associated registers**

None

**Extension registers**

ESTCHGS2

**Associated logs**

None

**PASTHRE****Register type**

Peg

**Description**

PASTHRE is pegged when an OAP Pass Through Error Response is sent to a service node.

To test this register, send this OAP message and verify that the register is pegged.

SOC OSAN0003 must be on for this OAP message.

**Associated registers**

None

**Extension registers**

PASTHRE2

**Associated logs**

None

**PASTHRQ****Register type**

Peg

**Description**

PASTHRQ is pegged when an OAP Pass Through Request is received.

To test this register, receive this OAP message and verify that the register is pegged.

SOC OSAN0003 must be on for this OAP message.

**Associated registers**

None

**Extension registers**

PASTHRQ2

**Associated logs**

None

**PASTHRS****Register type**

Peg

**Description**

PASTHRS is pegged when an OAP Pass Through Success Response is sent to a service node.

To test this register, send this OAP message and verify that the register is pegged.

SOC OSAN0003 must be on for this OAP message.

**Associated registers**

None

**Extension registers**

PASTHRS2

**Associated logs**

None

**SACTINF****Register type**

Peg

**Description**

SACTINF is pegged when a Session Active Inform OAP message is received from an OSSAIN service node.

**Associated registers**

None

**Extension registers**

SACTINF2

**Associated logs**

None

## OAPCP10

### Description

OM group Open Automated Protocol (OAP) Call Processing Group 10 (OAPCP10) contains registers that track call processing message operations and responses on an Operator Services Systems Advanced Intelligent Network (OSSAIN) session pool basis. A register is pegged each time the corresponding open automated protocol (OAP) message is sent or received by the DMS switch.

**Note:** For more information about OAP, refer to the OSSAIN Open Automated Protocol Specification, NIS: Q235-1.

The following table lists the key and info fields associated with OM group OAPCP10.

| Key field                                                                              | Info field |
|----------------------------------------------------------------------------------------|------------|
| SESNPLID (0-4096) - This field corresponds to the key field SESNPLID in table OASESNPL | None       |

### Related functional groups

The following functional groups are associated with OM group OAPCP10:

- OSSAIN
- OSAN0001

### Registers

The following table lists the registers associated with OM group OAPCP10 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPCP10 (Sheet 1 of 2)

| Register name           | Measures                                                        |
|-------------------------|-----------------------------------------------------------------|
| <a href="#">CBNSQRE</a> | Calling party billed number screening (BNS) query request error |
| <a href="#">CBNSQRQ</a> | Calling party billed number screening (BNS) query request       |

**Registers for OM group OAPCP10 (Sheet 2 of 2)**

| Register name           | Measures                                                          |
|-------------------------|-------------------------------------------------------------------|
| <a href="#">CBNSQRS</a> | Calling party billed number screening (BNS) query request success |
| <a href="#">RETANRE</a> | Return answer request error                                       |
| <a href="#">RETANRQ</a> | Return answer request                                             |
| <a href="#">RETANRS</a> | Return answer request success                                     |
| <a href="#">RNUPDRE</a> | Location routing number (LRN) update request error                |
| <a href="#">RNUPDRQ</a> | Location routing number (LRN) update request                      |
| <a href="#">RNUPDRS</a> | Location routing number (LRN) update request success              |
| <a href="#">SMSRQ</a>   | Number of SMS request operation messages received                 |
| <a href="#">SMSRE</a>   | Number of SMS error response messages sent to the SN              |
| <a href="#">SMSRS</a>   | Number of SMS success response messages sent to the SN            |

**CBNSQRE****Register type**

Peg

**Description**

CBMSQRE is pegged when an error is received on an OAP operation from a service node to perform a BNS query on the calling number. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CBNSQRE2

**Associated logs**

None

**CBNSQRQ****Register type**

Peg

**Description**

CBNSQRQ is pegged when a request is received in an OAP operation from a service node to perform a BNS query on the calling number. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CBNSQRQ2

**Associated logs**

None

**CBNSQRS****Register type**

Peg

**Description**

CBNSQRS is pegged when an OAP operation from a service node for a BNS query on the calling number is successfully processed. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

CBNSQRS2

**Associated logs**

None

**RETANRE****Register type**

Peg

**Description**

RETANRE is pegged when an error is received in a Return Answer operation from a service node. To test this register, receive the message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

RETANRE2

**Associated logs**

None

**RETANRQ****Register type**

Peg

**Description**

RETANRQ is pegged when the Return Answer operation is received from a service node. To test this register, receive the message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

RETANRQ2

**Associated logs**

None

**RETANRS****Register type**

Peg

**Description**

RETANRS is pegged when the Return Answer operation is received successfully from a service node. To test this register, receive the message associated with this register and verify that the register is pegged.



**Associated registers**

None

**Extension registers**

RETANRS2

**Associated logs**

None

**RNUPDRE****Register type**

Peg

**Description**

RNUPDRE is pegged when an error is received on an OAP operation to assign an LRN as requested by a service node. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

RNUPDRE2

**Associated logs**

None

**RNUPDRQ****Register type**

Peg

**Description**

RNUPDRQ is pegged when the Assign LRN operation is received from a service node. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

RNUPDRQ2

**Associated logs**

None

**RNUPDRS****Register type**

Peg

**Description**

RNUPDRS is pegged when an OAP operation to assign an LRN is successfully processed from a service node. To test this register, receive the OAP message associated with this register and verify that the register is pegged.

**Associated registers**

None

**Extension registers**

RNUPDRS2

**Associated logs**

None

**SMSRQ****Register type**

Peg

**Description**

SMSRQ records the number of SMS request operation messages received.

**Associated registers**

None

**Extension registers**

SMSRQ2

**Associated logs**

None

**SMSRE****Register type**

Peg

**Description**

SMSRE records the number of SMS error response messages sent to the SN.

**Associated registers**

None

**Extension registers**

SMSRE2

**Associated logs**

None

**SMSRS****Register type**

Peg

**Description**

SMSRS records the number of SMS success response messages sent to the SN.

**Associated registers**

None

**Extension registers**

SMSRS2

**Associated logs**

None

## OAPMERRN

### Description

OM group Open Automated Protocol (OAP) Message Error - Node (OAPMERRN) contains a register for the different types of errors that OAP messages can have. Each register in OM group OAPMERRN is pegged on a per node basis (OAP Node Maintenance class messages).

The following table lists the key and info fields associated with OM group OAPMERRN:

| Key field                                      | Info field |
|------------------------------------------------|------------|
| NODEID {0 to 96}: Key field for table OANODINV | None       |

### Related functional groups

ENSV Enhanced Services, ENSV0001 is associated with OM group OAPMERRN.

### Registers

The following table lists the registers associated with OM group OAPMERRN and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPMERRN (Sheet 1 of 2)

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">NERRORER</a> | Node Return Error Error          |
| <a href="#">NINVDFD</a>  | Node Invalid Field Value         |
| <a href="#">NINVKER</a>  | Node Invoke Error                |
| <a href="#">NINVOPHD</a> | Node Invalid Operation Header ID |
| <a href="#">NMSNGDB</a>  | Node Missing Data Block          |
| <a href="#">NREJCTE</a>  | Node Reject Error                |
| <a href="#">NRESLTER</a> | Node Return Result Error         |

**Registers for OM group OAPMERRN (Sheet 2 of 2)**

| Register name           | Measures                  |
|-------------------------|---------------------------|
| <a href="#">NUNKNDB</a> | Node Unknown Data Block   |
| <a href="#">NUNKNOP</a> | Node Unknown Operation ID |

**NERRORER****Register type**

Peg

**Description**

NERRORER is pegged each time the switch receives a node maintenance message which has an operation ID that the switch does not recognize.

**Note:** For test case(s), receive an error response with an error for a node maintenance request.

**Associated registers**

None

**Extension registers**

NERRORE2

**Associated logs**

OAP600

**NINVDFD****Register type**

Peg

**Description**

NINVDFD is pegged each time the switch receives a node maintenance message which has a field with an invalid value.

**Note:** For test case(s), receive an OAP for a node maintenance message with an invalid value in a field.

**Associated registers**

None

**Extension registers**

NINVDFD2

**Associated logs**

OAP600

**NINVKER****Register type**

Peg

**Description**

NINVKER is pegged each time the switch receives an invalid value in a field and in a node maintenance Invoke Operation Header.

**Note:** Currently, this register is not testable. Register NINVKER may be pegged in a future release.

**Associated registers**

None

**Extension registers**

NINVKER2

**Associated logs**

OAP600

**NINVOPHD****Register type**

Peg

**Description**

NINVOPHD is pegged each time the switch receives a node maintenance message which has an invalid Operation Header ID.

**Note:** For test case(s), receive a node maintenance message from a service node with an invalid operation header ID.

**Associated registers**

None

**Extension registers**

NINVOPH2

**Associated logs**

OAP600

**NMSGDB****Register type**

Peg

**Description**

NMSGDB is pegged each time the switch receives a node maintenance operation or response with a missing mandatory data block.

**Note:** For test case(s), receive a node maintenance response with a missing mandatory data block.

**Associated registers**

None

**Extension registers**

NMSGDB2

**Associated logs**

OAP602

**NREJCTE****Register type**

Peg

**Description**

NREJCTE is pegged each time the switch receives an invalid value in the Reject Operation Header for a node maintenance message.

**Note:** For test case(s), send a message with an invalid operation ID to the service node.

**Associated registers**

None

**Extension registers**

NREJCTE2

**Associated logs**

OAP600

**NRESLTER****Register type**

Peg

**Description**

NRESLTER is pegged each time the switch receives an invalid value in a node maintenance Return Result Operation Header.

**Note:** For test case(s), receive a success response with an invalid value in a field, in the Return Result Operation Header for a node maintenance request.

**Associated registers**

None

**Extension registers**

NRESLTE2

**Associated logs**

OAP600

**NUNKNDB****Register type**

Peg

**Description**

NUNKNDB is pegged each time the switch receives an operation or response with an unknown data block from a service node.

**Note:** For test case(s), receive a node maintenance message response with an unknown data block.

**Associated registers**

None

**Extension registers**

NUNKNDB2

**Associated logs**

OAP602

**NUNKNOP****Register type**

Peg



**Description**

NUNKNOP is pegged each time the switch receives a node maintenance message which has an operation ID that the switch does not recognize.

**Note:** Currently, this register is not testable. Register NINVKER may be pegged in a future release.

**Associated registers**

None

**Extension registers**

NUNKNOP2

**Associated logs**

OAP601

## OAPMERRS

### Description

OM group Open Automated Protocol (OAP) Message Error - Session Pool (OAPMERRS) contains a register for the different types of errors that OAP messages can have. Each register in OM group OAPMERRS is pegged on a per session pool basis (for example, OAP Call Processing class and OAP Session Pool Maintenance class messages).

The following table lists the key and info fields associated with OM group OAPMERRS:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services, ENSV0001 is associated with OM group OAPMERRS.

### Registers

The following table lists the registers associated with OM group OAPMERRS and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPMERRS (Sheet 1 of 2)

| Register name            | Measures                    |
|--------------------------|-----------------------------|
| <a href="#">INVDFLD</a>  | Invalid Field Value         |
| <a href="#">INVKERR</a>  | Invoke Error                |
| <a href="#">INVLCLL</a>  | Invalid Call ID             |
| <a href="#">INVLFN</a>   | Invalid Function ID         |
| <a href="#">INVLOPHD</a> | Invalid Operation Header ID |
| <a href="#">MISNGDB</a>  | Missing Data Block          |
| <a href="#">REJECTER</a> | Reject Error                |

**Registers for OM group OAPMERRS (Sheet 2 of 2)**

| Register name            | Measures               |
|--------------------------|------------------------|
| <a href="#">RERORER</a>  | Return Error Error     |
| <a href="#">RRESLTER</a> | Return Result Error    |
| <a href="#">SEQERRH</a>  | Out of Sequence - High |
| <a href="#">SEQERRL</a>  | Out of Sequence - Low  |
| <a href="#">UNKWNDB</a>  | Unknown Data Block     |
| <a href="#">UNKWNOP</a>  | Unknown Operation ID   |

**INVDFLD****Register type**

Peg

**Description**

INVDFLD is pegged each time the switch receives a call processing or session pool maintenance message which has a field with an invalid value.

**Note:** For test case(s), receive an OAP message with an invalid value in a field.

**Associated registers**

None

**Extension registers**

INVDFLD2

**Associated logs**

OAP600

**INVKERR****Register type**

Peg

**Description**

INVKERR is pegged each time the switch receives an invalid value in the Invoke Operation Header of a call processing or session pool maintenance message.

**Note:** For test case(s), receive an operation request from a service node that has an invalid value in the Invoke Operation Header.

**Associated registers**

None

**Extension registers**

INVKERR2

**Associated logs**

OAP600

**INVLCLL****Register type**

Peg

**Description**

INVLCLL is pegged each time the switch receives a call processing class message with an invalid call ID. Session pool maintenance does not peg this register.

**Note:** For test case(s), receive a message from a service node with an invalid call ID.

**Associated registers**

None

**Extension registers**

INVLCLL2

**Associated logs**

None

**INVLFN****Register type**

Peg

**Description**

INVLFN is pegged each time the switch receives a call processing class message with an invalid function ID. The function ID is either out

of the valid range or inconsistent with prior messages sent for this session. Session pool maintenance does not peg this register.

**Note:** For test case(s), receive a message from a service node with an invalid function ID.

**Associated registers**

None

**Extension registers**

INVLFN2

**Associated logs**

OAP600

**INVLOPHD**

**Register type**

Peg

**Description**

INVLOPHD is pegged each time the switch receives a message which has an invalid operation header ID.

**Note:** For test case(s), receive a message from a service node with an invalid operation header ID.

**Associated registers**

None

**Extension registers**

RERRORER2

**Associated logs**

OAP600

**MISNGDB**

**Register type**

Peg

**Description**

MISNGDB is pegged each time the switch receives a call processing, session pool maintenance message operation, or response with a missing mandatory data block.

**Note:** For test case(s), receive an operation request with a missing mandatory data block.

**Associated registers**

None

**Extension registers**

MISNGDB2

**Associated logs**

OAP602

**REJECTER****Register type**

Peg

**Description**

REJECTER is pegged each time the switch receives an invalid value in the Reject Operation Header of a call processing or session pool maintenance message.

**Note:** For test case(s), send a message with an invalid operation ID to the service node.

**Associated registers**

None

**Extension registers**

REJECTE2

**Associated logs**

OAP600

**RERRORER****Register type**

Peg

**Description**

RERRORER is pegged each time the switch receives an invalid value in the Return Error Operation Header of a call processing or session pool maintenance message.

**Note:** For test case(s), receive an error response with an error for a session pool maintenance request.

**Associated registers**

None

**Extension registers**

NUNKNDB2

**Associated logs**

OAP600

**RRESLTER****Register type**

Peg

**Description**

RRESLTER is pegged each time the switch receives an invalid value in the Return Result Operation Header of a call processing or session pool maintenance message.

**Note:** For test case(s), receive a success response with an error for a session pool maintenance request.

**Associated registers**

None

**Extension registers**

RRESLTE2

**Associated logs**

OAP600

**SEQERRH****Register type**

Peg

**Description**

SEQERRH is pegged each time the switch receives a call processing class message that is out of sequence and the sequence number is higher than what the switch is expecting. Session pool maintenance does not peg this register.

**Note:** For test case(s), receive an out-of-sequence message from a service node with a higher sequence number than what the switch is expecting.

**Associated registers**

None

**Extension registers**

SEQERRH2

**Associated logs**

OAP600

**SEQERRL****Register type**

Peg

**Description**

SEQERRL is pegged each time the switch receives a call processing class message that is out of sequence and the sequence number is lower than what the switch is expecting. Session pool maintenance does not peg this register.

**Note:** For test case(s), receive an out-of-sequence message from a service node with a lower sequence number than what the switch is expecting.

**Associated registers**

None

**Extension registers**

SEQERRL2

**Associated logs**

None

**UNKWNDB****Register type**

Peg

**Description**

UNKWNDB is pegged each time the switch receives a call processing, session pool maintenance message operation, or response with an unknown data block from a service node.

**Note:** For test case(s), receive an operation request with an unknown data block.

**Associated registers**

None

**Extension registers**

UNKWNDB2

**Associated logs**

OAP602



**UNKWNOP****Register type**

Peg

**Description**

UNKWNOP is pegged each time the switch receives a call processing or session pool maintenance message that has an operation ID the switch does not recognize.

**Note:** For test case(s), receive an operation request from a service node with an unknown operation ID.

**Associated registers**

None

**Extension registers**

UNKWNOP2

**Associated logs**

OAP601

## OAPMTYPN

### Description

OM group Open Automated Protocol (OAP) Message Type - Node (OAPMTYPN) contains a register for each incoming and outgoing OAP message type. OM group OAPMTYPN registers are pegged for node based messages (for example, Node Maintenance class messages) on a per node basis.

The following table lists the key and info fields associated with OM group OAPMTYPN:

| Key field                                      | Info field |
|------------------------------------------------|------------|
| NODEID {0 to 96}: Key field for table OANODINV | None       |

### Related functional groups

ENSV Enhanced Services, ENSV0001 is associated with OM group OAPMTYPN.

### Registers

The following table lists the registers associated with OM group OAPMTYPN and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPMTYPN

| Register name           | Measures             |
|-------------------------|----------------------|
| <a href="#">NIERROR</a> | Node Incoming Error  |
| <a href="#">NIINVOK</a> | Node Incoming Invoke |
| <a href="#">NIREJCT</a> | Node Incoming Reject |
| <a href="#">NIRESLT</a> | Node Incoming Result |
| <a href="#">NOERROR</a> | Node Outgoing Error  |
| <a href="#">NOINVOK</a> | Node Outgoing Invoke |
| <a href="#">NOREJCT</a> | Node Outgoing Reject |
| <a href="#">NORESLT</a> | Node Outgoing Result |

**NIERROR****Register type**

Peg

**Description**

NIERROR is pegged each time the switch receives a node maintenance error response on a per node level basis.

**Note:** For test case(s), receive a node maintenance response for a busy request (for example, a busy error response).

**Associated registers**

None

**Extension registers**

NIERROR2

**Associated logs**

None

**NIINVOK****Register type**

Peg

**Description**

NIINVOK is pegged each time the switch receives an incoming node maintenance inform or request operation on a per node level basis.

**Note:** Currently, this register is not testable. Register NINVOK may be pegged in a future release.

**Associated registers**

None

**Extension registers**

NIINVOK2

**Associated logs**

None

**NIREJCT****Register type**

Peg

**Description**

NIREJCT is pegged each time the switch receives a node maintenance protocol violation on a per node level basis.

**Note:** For test case(s), send a message to a service node with an invalid operation ID.

**Associated registers**

None

**Extension registers**

NIREJCT2

**Associated logs**

None

**NIRESLT****Register type**

Peg

**Description**

NIRESLT is pegged each time the switch receives a node maintenance success response on a per node level basis.

**Note:** For test case(s), receive a response for a node maintenance request (for example, a busy success response).

**Associated registers**

None

**Extension registers**

NIRESLT2

**Associated logs**

None

**NOERROR****Register type**

Peg

**Description**

NOERROR is pegged each time the switch sends an error response for a node maintenance operation request on a per node level basis.

**Note:** Currently, this register is not testable. Register NOERROR may be pegged in a future release.

**Associated registers**

None

**Extension registers**

NOERROR2

**Associated logs**

None

**NOINVOK****Register type**

Peg

**Description**

NOINVOK is pegged each time the switch sends a node maintenance inform or request operation on a node level basis.

**Note:** For test case(s), send a maintenance request to a node (for example, a node busy).

**Associated registers**

None

**Extension registers**

NOINVOK2

**Associated logs**

None

**NOREJCT****Register type**

Peg

**Description**

NOREJCT is pegged each time the switch sends a node maintenance protocol violation on a per node level basis.

**Note:** For test case(s), receive a message with an invalid operation ID.

**Associated registers**

None

**Extension registers**

NOREJCT2

**Associated logs**

OAP600, OAP601, OAP602

**NORESLT****Register type**

Peg

**Description**

NORESLT is pegged each time the switch sends a success response for a node maintenance operation request on a per node level basis.

**Note:** Currently, this register is not testable. Register NOERROR may be pegged in a future release.

**Associated registers**

None

**Extension registers**

NORESLT2

**Associated logs**

None

## OAPMTYPS

### Description

OM group Open Automated Protocol (OAP) Message Type - Session Pool (OAPMTYPS) contains a register for each incoming and outgoing OAP message type. OM group OAPMTYPS registers are pegged for session pool based messages (for example, OAP Call Processing class and OAP Session Pool Maintenance class messages) on a per session pool basis.

The following table lists the key and info fields associated with OM group OAPMTYPS:

| Key field                                          | Info field |
|----------------------------------------------------|------------|
| SESNPLID {0 to 4094}: Key field for table OASESNPL | None       |

### Related functional groups

ENSV Enhanced Services, ENSV0001 is associated with OM group OAPMTYPS.

### Registers

The following table lists the registers associated with OM group OAPMTYPS and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OAPMTYPS (Sheet 1 of 2)

| Register name           | Measures        |
|-------------------------|-----------------|
| <a href="#">INERROR</a> | Incoming Error  |
| <a href="#">ININVOK</a> | Incoming Invoke |
| <a href="#">INREJCT</a> | Incoming Reject |
| <a href="#">INRESLT</a> | Incoming Result |
| <a href="#">OGERROR</a> | Outgoing Error  |
| <a href="#">OGINVOK</a> | Outgoing Invoke |

**Registers for OM group OAPMTYPS (Sheet 2 of 2)**

| Register name           | Measures        |
|-------------------------|-----------------|
| <a href="#">OGREJCT</a> | Outgoing Reject |
| <a href="#">OGRESLT</a> | Outgoing Result |

**INERROR****Register type**

Peg

**Description**

INERROR is pegged each time the switch receives a call processing or session pool maintenance error response from a session pool.

**Note:** For test case(s), receive a response for a busy request (for example, a busy error response).

**Associated registers**

None

**Extension registers**

INERROR2

**Associated logs**

None

**ININVOK****Register type**

Peg

**Description**

ININVOK is pegged each time the switch receives an incoming call processing, session pool maintenance request, or inform operation from a session pool.

**Note:** For test case(s), make a call to a service node that requires the service node to request a voice connection.

**Associated registers**

None

**Extension registers**

ININVOK2



**Associated logs**

None

**INREJCT****Register type**

Peg

**Description**

INREJCT is pegged each time the switch receives a call processing or session pool maintenance protocol violation from a session pool.

**Note:** For test case(s), send a message to a service node with an invalid function id.

**Associated registers**

None

**Extension registers**

INREJCT2

**Associated logs**

None

**INRESLT****Register type**

Peg

**Description**

INRESLT is pegged each time the switch receives a call processing or session pool maintenance success response from a session pool.

**Note:** For test case(s), receive a session pool response for a maintenance request (for example, a busy success response).

**Associated registers**

None

**Extension registers**

INRESLT2

**Associated logs**

None

**OGERROR****Register type**

Peg

**Description**

OGERROR is pegged each time the switch sends a call processing or session pool maintenance error response for an operation request to a session pool.

**Note:** For test case(s), make a call to a service node for a function that requires a voice connection. Busy all the voice links. The switch will detect that all voice links are unavailable. It will then send an error response to the service node.

**Associated registers**

None

**Extension registers**

OGERROR2

**Associated logs**

None

**OGINVOK****Register type**

Peg

**Description**

OGINVOK is pegged each time the switch sends a call processing, session pool maintenance request, or inform operation to a session pool.

**Note:** For test case(s), send a session pool maintenance request to a session pool (for example, a session pool busy).

**Associated registers**

None

**Extension registers**

OGINVOK2

**Associated logs**

None

**OGREJCT****Register type**

Peg

**Description**

OGREJCT is pegged each time the switch sends a protocol violation to a session pool.

**Note:** For test case(s), receive a message with an invalid operation ID.

**Associated registers**

None

**Extension registers**

OGREJCT2

**Associated logs**

OAP600, OAP601, OAP602

**OGRESLT****Register type**

Peg

**Description**

OGRESLT is pegged each time the switch sends a call processing or session pool maintenance success response for an operation request to a session pool.

**Note:** For test case(s), make a call to a service node which requires the service node to successfully request a voice connection.

**Associated registers**

None

**Extension registers**

OGRESLT2

**Associated logs**

None

## OAPNMIS

### Description

Operator services advanced intelligent network (OSSAIN) advanced protocol (OAP) node management information system This OM group is pegged for management information system (MIS) node class message types on a per node level basis.

OM group OAPNMIS provides up to 768 tuples per office. A tuple is added to this OM group for each NODEID datafilled in table OANODNAM.

| Key field | Info field |
|-----------|------------|
| NODEID    | none       |

- Key field:  
NODEID {0 to 767}: Key field for table OANODNAM

### Related functional groups

Functional group OSSAIN (OSAN0001) is associated with OM group OSACCP2.

### Registers

The following table lists the registers associated with OM group OAPNMIS and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OAPNMIS

| Register name           | Measures                             |
|-------------------------|--------------------------------------|
| <a href="#">MISOAIN</a> | Management information system OSSAIN |

#### MISOAIN

##### Register type

Peg

##### Description

Management information system OSSAIN (MISOAIN) Register MISOAIN is pegged each time the switch sends an MIS data buffer to a node.

##### Associated registers

There are no associated registers.

**Extension registers**

MISOAIN2

**Associated logs**

There are no associated logs.

## OAPNMTC

### Description

OM group Open Automated Protocol Node Maintenance Operations and Responses (OAPNMTC) contains a register for each call processing and non-call processing operation and response message defined in the OAP protocol. The registers track usage of the operations and responses. The registers are pegged on a per-session pool basis for call processing and session pool operations, and on a per-node basis for node maintenance operations.

The following table lists the key and info fields associated with OM group OAPNMTC. The group provides one tuple for each key value.

| Key field                                         | Info field                                           |
|---------------------------------------------------|------------------------------------------------------|
| NODEID {0 to 767}<br>Key field for table OANODNAM | OAP_NODE_INDEX_REGISTER<br>INFO (max. 16 characters) |

### Related functional groups

Functional group ENSV Enhanced Services (ENSV0001) is associated with OM group OAPNMTC.

### Registers

The following table lists the registers associated with OM group OAPNMTC and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OAPNMTC (Sheet 1 of 2)

| Register name           | Measures                                    |
|-------------------------|---------------------------------------------|
| <a href="#">NDALARM</a> | Node alarm operation                        |
| <a href="#">NDEAUDE</a> | Node audit error response                   |
| <a href="#">NDEAUDS</a> | Node audit success response                 |
| <a href="#">NDEBSYE</a> | Node busy error response                    |
| <a href="#">NDEBSYS</a> | Node busy success response                  |
| <a href="#">NDERTSE</a> | Node return-to-service (RTS) error response |
| <a href="#">NDERTSS</a> | Node RTS success response                   |

**Registers for OM group OAPNMTC (Sheet 2 of 2)**

| Register name           | Measures                              |
|-------------------------|---------------------------------------|
| <a href="#">NDETSTE</a> | Node test error response              |
| <a href="#">NDETSTS</a> | Node test success response            |
| <a href="#">NODEAUD</a> | Node audit request                    |
| <a href="#">NODEBSY</a> | Node busy request                     |
| <a href="#">NODERTS</a> | Node RTS request                      |
| <a href="#">NODETST</a> | Node test request                     |
| <a href="#">NDLOG</a>   | Node log report operation             |
| <a href="#">NODECON</a> | Node connectivity test                |
| <a href="#">NDECONS</a> | Node connectivity success response    |
| <a href="#">NDECONE</a> | Node connectivity test error response |

**NDALARM****Register type**

Peg

**Description**

NDALARM counts log report operations received for a given service node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDEAUDE****Register type**

Peg

**Description**

NDEAUDE counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDEAUDE2

**Associated logs**

None

**NDEAUDS****Register type**

Peg

**Description**

NDEAUDS counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDEAUDS2

**Associated logs**

None

**NDEBSYE****Register type**

Peg

**Description**

NDEBSYE counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.



**Associated registers**

None

**Extension registers**

NDEBSYE2

**Associated logs**

None

**NDEBSYS****Register type**

Peg

**Description**

NDEBSYS counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDEBSYS2

**Associated logs**

None

**NDERTSE****Register type**

Peg

**Description**

NDERTSE counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDERTSE2

**Associated logs**

None

**NDERTSS****Register type**

Peg

**Description**

NDERTSS counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDERTSS2

**Associated logs**

None

**NDETSTE****Register type**

Peg

**Description**

NDETSTE counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDETSTE2

**Associated logs**

None

**NDETSTS****Register type**

Peg

**Description**

NDETSTS counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NDETSTS2

**Associated logs**

None

**NODEAUD****Register type**

Peg

**Description**

NODEAUD counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NODEAUD2

**Associated logs**

None

**NODEBSY****Register type**

Peg

**Description**

NODEBSY counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NODEBSY2

**Associated logs**

None

**NODERTS****Register type**

Peg

**Description**

NODERTS counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NODERTS2

**Associated logs**

None

**NODETST****Register type**

Peg

**Description**

NODETST counts the number of times the switch sends or receives the corresponding node maintenance operation or response.

**Note:** For test case(s), make a call that requires the corresponding node maintenance operation or response.

**Associated registers**

None

**Extension registers**

NODETST2

**Associated logs**

None

**NDLOG****Register type**

Peg

**Description**

NDLOG counts the number of alarm operations received for the given service node.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NODECON****Register type**

Peg

**Description**

NODECON counts the number of times the switch sends or receives a node connectivity test or response.

**Associated registers**[NDECONS](#), [NDECON2](#)**Extension registers**

NODECON2

**Associated logs**

None

**NDECONS****Register type**

Peg

**Description**

NDECONS counts the number of times the switch sends or receives a node connectivity test success response.

**Associated registers**  
[NODECON](#), [NDECONE](#)

**Extension registers**  
NDECONS2

**Associated logs**  
None

## **NDECONE**

**Register type**  
Peg

**Description**  
NDECONE counts the number of times the switch sends or receives a node connectivity test error.

**Associated registers**  
[NODECON](#), [NDECONS](#)

**Extension registers**  
NDECONE2

**Associated logs**  
None

## OAPSPMTC

### Description

Open Automated Protocol (OAP) Session Pool Maintenance Operations and Responses

OAPSPMTC contains a register for each non-call processing operation and response message defined in the OAP protocol. The purpose of the registers in this OM group is to track usage of the operations and responses. These OM registers are pegged on a per session pool basis for non-call processing and session pool operations.

OM group OAPSPMTC provides up to 4095 tuples per office.

| Key field   | Info field                |
|-------------|---------------------------|
| OASVNDCP... | OAP_SP_INDEX_REGISTERINFO |

- Key field:  
OASVNDCP can be indexed by either of the following:
  - SESNPLID {0 to 4094}: Key field for table OASESNPL.
  - SESNPLNM: Name associated with SESNPLID.
- Info field:  
OAP\_SP\_INDEX\_REGISTERINFO - This name can be up to 16 characters long.

### Related functional groups

Functional group ENSV Enhanced Services (ENSV0001) is associated with OM group OAPSPMTC.

### Registers

The following table lists the registers associated with OM group OAPSPMTC and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OAPSPMTC

| Register name           | Measures                          |
|-------------------------|-----------------------------------|
| <a href="#">SPALARM</a> | Register Alarm Operation          |
| <a href="#">SPAUDIT</a> | Session Pool Audit Request        |
| <a href="#">SPAUDTE</a> | Session Pool Audit Error Response |

**Registers for OM group OAPSPMTC**

| Register name            | Measures                            |
|--------------------------|-------------------------------------|
| <a href="#">SPAUDTS</a>  | Session Pool Audit Success Response |
| <a href="#">SPBUSY</a>   | Session Pool Busy Request           |
| <a href="#">SPBUSYE</a>  | Session Pool Busy Error Response    |
| <a href="#">SPBUSYS</a>  | Session Pool Busy Success Response  |
| <a href="#">SPCH</a>     | Register Throttle Operation         |
| <a href="#">SPCHE</a>    | Register Throttle Operation Error   |
| <a href="#">SPCHS</a>    | Register Throttle Operation Success |
| <a href="#">SPDRAIN</a>  | Register Drain Operation            |
| <a href="#">SPLOG</a>    | Register Log Report Operation       |
| <a href="#">SPRTS</a>    | Session Pool RTS Request            |
| <a href="#">SPRTSE</a>   | Session Pool RTS Error Response     |
| <a href="#">SPRTSS</a>   | Session Pool RTS Success Response   |
| <a href="#">SPSTATE</a>  | Session Pool State Inform           |
| <a href="#">SPTTEST</a>  | Session Pool Test Request           |
| <a href="#">SPTTESTE</a> | Session Pool Test Error Response    |
| <a href="#">SPTTESTS</a> | Session Pool Test Success Response  |

**SPALARM****Register type**

Peg

**Description**

Register Alarm Operation

This register pegs the number of alarm operations for the given session pool.

**Associated registers**

There are no associated registers.



**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPAUDIT****Register type**

Peg

**Description**

Session Pool Audit Request

This register is pegged each time the audit request is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPAUDIT2

**Associated logs**

There are no associated logs.

**SPAUDTE****Register type**

Peg

**Description**

Session Pool Audit Error Response

This register is pegged each time a session pool audit error response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPAUDTE2

**Associated logs**

There are no associated logs.

**SPAUDTS****Register type**

Peg

**Description**

Session Pool Audit Success Response

This register is pegged each time an audit success response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPAUDTS2

**Associated logs**

There are no associated logs.

**SPBUSY****Register type**

Peg

**Description**

Session Pool Busy Request This register is pegged each time a busy request is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPBUSY2

**Associated logs**

There are no associated logs.

**SPBUSYE****Register type**

Peg

**Description**

Session Pool Busy Error Response

This register is pegged each time a busy error response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPBUSYE2

**Associated logs**

There are no associated logs.

**SPBUSYS****Register type**

Peg

**Description**

Session Pool Busy Success Response

This register is pegged each time a busy success response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPBUSYS2

**Associated logs**

There are no associated logs.

**SPCH****Register type**

Peg

**Description**

Register Throttle Operation

This register pegs the number of requests to change the number of active sessions in the given session pool.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPCHE****Register type**

Peg

**Description**

Register Throttle Operation Error

This register pegs the number of throttle requests that sends an error response back to the requestor.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPCHS****Register type**

Peg

**Description**

Register Throttle Operation Success

This register pegs the number of throttle requests that sends a success response back to the requestor.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPDRAIN****Register type**

Peg

**Description**

Register Drain Operation

This register pegs the number of drain operations for the given session pool.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPLOG****Register type**

Peg

**Description**

Register Log Report Operation

This register pegs the number of log report operations for the given session pool.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPRTS****Register type**

Peg

**Description**

Session Pool RTS Request

This register is pegged each time an RTS request is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPRTS2

**Associated logs**

There are no associated logs.

**SPRTSE****Register type**

Peg

**Description**

Session Pool RTS Error Response

This register is pegged each time an RTS error response is sent or received by the switch.

**Associated registers**

SPRTSE2

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPRTSS****Register type**

Peg

**Description**

Session Pool RTS Success Response

This register is pegged each time an RTS success response is sent or received by the switch.

**Associated registers**

SPRTSS2

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SPSTATE****Register type**

Peg

**Description**

Session Pool State Inform

This register is pegged each time the switch sends an OAP Session Pool State Inform message to the service node.

**Associated registers**

There are no associated registers.

**Extension registers**

SPSTATE2

**Associated logs**

There are no associated logs.

**SPTTEST****Register type**

Peg

**Description**

Session Pool Test Request

This register is pegged each time a Test request is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPTTEST2

**Associated logs**

There are no associated logs.

**SPTESTE****Register type**

Peg

**Description**

Session Pool Test Error Response

This register is pegged each time a Test error response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPTESTE2

**Associated logs**

There are no associated logs.

**SPTTESTS****Register type**

Peg

**Description**

Session Pool Test Success Response

This register is pegged each time a Test success response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPTTESTS2

**Associated logs**

There are no associated logs.



## OASNPLDC

### Description

Operator Services System Advanced Intelligent Network (OSSAIN)  
Session Pool Data Communications

OASNPLDC (OSSAIN Session Pool Data Communications) is created for data communications operational measurements. The following OM groups are also created for data communications operational measurements:

- OADATCOM - OSSAIN Data Communications
- OANODEDC - OSSAIN Node Data Communications

OM group OASNPLDC provides peg counts for OSSAIN data communications messaging events on a per session pool basis. It provides counts for the total number of messages sent from the CM to each session pool and the total number of messages received by the CM from each session pool. Counts of messages are broken down into successful and failure counts.

OM group OASNPLDC provides one tuple for each key.

| Key field | Info field                   |
|-----------|------------------------------|
| SESSPLID  | OSSAIN_SESNPL_DATACOM_OMINFO |

- Key field:  
SESSPLID {0 - 4094}: Key field from table OASESNPL

### Related functional groups

Functional group ENSV Enhanced Services (ENSV0001) is associated with OM group OASNPLDC.

## Registers

The following table lists the registers associated with OM group OASNPLDC and what they measure. For a description of a register, click on the register name.

### Registers for OM group OASNPLDC

| Register name            | Measures                                              |
|--------------------------|-------------------------------------------------------|
| <a href="#">OSMSGRCV</a> | OSSAIN message received per session pool              |
| <a href="#">OSMSGRFL</a> | OSSAIN message receive failure per session pool       |
| <a href="#">OSMSGRSC</a> | OSSAIN message receive success per session pool       |
| <a href="#">OSMSGSFL</a> | OSSAIN message send failure per session pool          |
| <a href="#">OSMSGSND</a> | OSSAIN message send requested per session pool        |
| <a href="#">OSMSGSSC</a> | OSSAIN message send success per session pool          |
| <a href="#">OSRCRTFL</a> | OSSAIN message receive route failure per session pool |
| <a href="#">OSSNRTFL</a> | OSSAIN message receive route failure per session pool |

#### OSMSGRCV

##### Register type

Peg

##### Description

OSSAIN message received per session pool

This register is pegged for a specific session pool each time an incoming message, originating from an external node, is received from that session pool. This includes both call processing and maintenance messages.

**Note:** This register can be validated on a per session pool basis by adding the message receive success register and the message receive failure register that apply to the node of interest.

$\text{OSMSGRCV} = \text{OSMSGRSC} + \text{OSMSGRFL}$

**Associated registers**

[OSMSGRSC](#) and [OSMSGRFL](#)

**Extension registers**

OSMSGRC2

**Associated logs**

There are no associated logs.

**OSMSGRFL**

**Register type**

Peg

**Description**

OSSAIN message receive failure per session pool

This register is pegged for a specific session pool each time data communications encounters an error while attempting to forward a message originated from that session pool to the destination DMS process. This can be caused by a failure in the DMS internal messaging system or data transport interface. This register is also pegged for reasons indicated by register OSRCRTFL.

**Note:** The validation formula for this registers follows:

$\text{OSMSGRFL} = \text{OSMSGRC} - \text{OSMSGRSC}$   
 $\text{OSMSGRFL} \geq \text{OSRCRTFL}$

**Associated registers**

[OSMSGRCV](#), [OSMSGRSC](#), and [OSRCRTFL](#)

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**OSMSGRSC**

**Register type**

Peg

**Description**

OSSAIN message receive success per session pool

This register is pegged for a specific session pool when the data communications software of the CM is able to successfully process an incoming message from the session pool.

**Note:** The validation formula for this registers follows:

$$\text{OSMSGRSC} = \text{OSMSGRC} - \text{OSMSGRFL}$$

**Associated registers**

[OSMSGRCV](#) and [OSMSGRFL](#)

**Extension registers**

OSMSGRS2

**Associated logs**

There are no associated logs.

**OSMSGSFL**

**Register type**

Peg

**Description**

OSSAIN message send failure per session pool

This register is pegged for a specific session pool each time data communications encounters an error while attempting to send an outgoing message to the session pool. This can be caused by a transport layer failure. This register is also pegged for reasons indicated by register OSSNRTFL.

**Note:** The validation formula for this registers follows:

$$\begin{aligned}\text{OSMSGSFL} &= \text{OSMSGSND} - \text{OSMSGSSC} \\ \text{OSMSGSFL} &\geq \text{OSSNRTFL}\end{aligned}$$

**Associated registers**

[OSMSGSND](#), [OSMSGSSC](#), and [OSSNRTFL](#)

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**OSMSGSND**

**Register type**

Peg

**Description**

OSSAIN message send requested per session pool

This register is pegged for a specific session pool each time the data communications software is requested to send a message. This includes requests from call processes and maintenance processes.

**Note:** This register can be validated on a per session pool basis by adding the message send success register and the message send failure register that apply to the session pool of interest.

$$\text{OSMSGSDND} = \text{OSMSGSSC} + \text{OSMSGDFL}$$

**Associated registers**

[OSMSGSSC](#) and [OSMSGDFL](#)

**Extension registers**

OSMSGSDN2

**Associated logs**

There are no associated logs.

**OSMSGSSC****Register type**

Peg

**Description**

OSSAIN message send success per session pool

This register is pegged for a specific session pool when the data communications software of the CM is able to successfully process an outgoing message destined for that session pool. Note that OSSAIN uses non-guaranteed messaging. Pegging this register does not indicate that the message actually arrived at the destination session pool.

**Note:** The validation formula for this registers follows:

$$\text{OSMSGSDND} \text{ and } \text{OSMSGDFL}$$

**Associated registers**

[OSMSGSDND](#) and [OSMSGDFL](#)

**Extension registers**

OSMSGSS2

**Associated logs**

There are no associated logs.

**OSRCRTFL****Register type**

Peg

**Description**

OSSAIN message receive route failure per session pool

This register is pegged for a specific session pool each time the data communications software is unable to determine the destination of a message originating from that session pool. This can be caused by a variety of reasons including:

- invalid protocol version

- invalid session identifier
- invalid network address
- invalid session pool state
- invalid node state
- pool/node identifier mis-match
- corrupted message

**Note:** The validation formula for this registers follows:

$$\text{OSRCRTFL} \leq \text{OSMSGRFL}$$
**Associated registers**

OSMSGRFL

**Extension registers**

There are no extension registers.

**Associated logs**

Log number: OAIN605 and OAIN606

**OSSNRTFL****Register type**

Peg

**Description**

OSSAIN message receive route failure per session pool

This register is pegged for a specific session pool each time the data communications software is unable to determine the destination of an outgoing message. This can be caused by the following reasons:

- invalid session identifier
- pool/node identifier mis-match
- corrupted message

**Note:** The validation formula for this registers follows:

OSSNRTFL <= OSMSGSL

#### **Associated registers**

[OSMSGSL](#)

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

There are no associated logs.

## OASNPOOL

### Description

#### Session Pool Inventory

This group provides the craftsperson with information regarding maintenance level activity on session pools datafilled in table OASESNPL. Specific information on maintenance detected and manual outages of a particular session pool is provided.

OMgroup OASNPOOL provides up to 4095 session pool tuples, one tuple per session pool datafilled in table OASESNPL.

| Key field | Info field        |
|-----------|-------------------|
| None      | session_pool_name |

- Info field:  
(SIXTEEN\_CHARS [16 Character Session Pool Name])

### Related functional groups

Functional group ENSV Enhanced Services (ENSV0001) is associated with OM group OASNPOOL.

### Registers

The following table lists the registers associated with OM group OASNPOOL and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OASNPOOL

| Register name           | Measures                                  |
|-------------------------|-------------------------------------------|
| <a href="#">RTSFAIL</a> | Session Pool Return-to-Service (RTS) Fail |
| <a href="#">SPCBSY</a>  | Session Pool C-Side Busy (CBSY)           |
| <a href="#">SPMANB</a>  | Session Pool MANB                         |
| <a href="#">SPSYSB</a>  | Session Pool SYSB                         |
| <a href="#">TSTFAIL</a> | Session Pool Test Fail                    |



## RTSFAIL

### Register type

Peg

### Description

Session Pool Return-to-Service (RTS) Fail

This register counts the number of times that a specific session pool failed to RTS—whether by audit or manual RTS.

**Note 1:** This register cannot be tested from the computing module (CM). Refer to service node session pool applications documentation to disable session pool audits which would cause the CM session pool audit to fail and the session pool to be marked system busy (SYSB) at the MAP.

**Note 2:** This test capability may not be provided by all session pool applications.

At the MAP, place the session pool in the manual busy (MANB) state and attempt to RTS the session pool by entering RTS. At the MAP, verify that the RTS fails, and that the session pool changes to the SYSB state. Also verify that an OAIN500 (DIAG FAIL) and OAIN502 (SYSB) log is produced, and that the RTSFAIL and SPSYSB OM registers are pegged.

### Associated registers

[SPSYSB](#)

### Extension registers

There are no extension registers.

### Associated logs

OAIN500

## SPCBSY

### Register type

Peg

### Description

Session Pool C-Side Busy (CBSY)

This register counts the number of times that a specific session pool went c-side busy.

**Note:** For test case(s), with the session pool in-service, busy the service node. Using the MAP, verify that the session pool goes CBSY

and SPCBSY and is pegged. Verify that an OAIN507 log is produced, in addition to a PM105 log. Verify that a minor alarm is generated.

Repeat above test case, but instead of setting the service node to the MANB state, MANB the Ethernet interface unit (EIU) interfaced to the service node. This action should cause the service node to change to the SYSB state. Again, using the MAP, verify that the session pool went CBSY, the SPCBSY OM register is pegged, an OAIN507 log is produced (in addition to a PM102 [SYSB] log), and a major alarm is generated.

**Associated registers**

INSSYSB (Count of node going from in-service to SYSB) or INSMANB (Count of node going from in-service to MANB).

**Extension registers**

There are no extension registers.

**Associated logs**

OAIN507

**SPMANB****Register type**

Peg

**Description**

Session Pool MANB

This register counts the number of times that a specific session pool went MANB.

**Note:** To test this register, at a MAP terminal, post a session pool and busy (BSY) it. Verify the OM count is incremented and an OAIN505 and an PM128(ISTB) log is produced, along with a minor alarm.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

OAIN505

**SPSYSB****Register type**

Peg

**Description**

Session Pool SYSB

This register counts the number of times that a specific session pool went SYSB.

**Note:** This register cannot be tested from the CM. Refer to service node session pool applications documentation to disable session pool audits that would cause the session pool audit to fail and the session pool to be marked SYSB at the MAP.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

OAIN502

**TSTFAIL****Register type**

Peg

**Description**

Session Pool Test Fail

This register counts the number of times that a specific session pool failed to successfully complete a diagnostic test.

**Note 1:** This register cannot be tested from the CM. Refer to service node session pool applications documentation to disable session pool audit responses that would cause the CM session pool audit to fail and the session pool to be marked SYSB at the MAP.

**Note 2:** This test capability may not be provided by all session pool applications. At the MAP, place the session pool in the MANB state and attempt to test the session pool by entering TST.

At the MAP, verify that the TST fails, and that the session pool stays MANB. Also verify that an OAIN500 (DIAG FAIL) log is produced and the TSTFAIL OM register is pegged.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## OASVNDCP

### Description

Operator Services Systems Advanced Intelligent Network (OSSAIN) Service Node Call Processing

OM group OASVNDCP provides peg counts for OSSAIN calls on a per session pool basis. It provides counts for all service node or OSAC call processing activities.

OMgroup OASVNDCP provides up to 4095 tuples per office. A tuple is added for each session pool defined in table OASESNPL.

| Key field   | Info field                  |
|-------------|-----------------------------|
| OASVNDCP... | OASVNDCP_INDEX_REGISTERINFO |

- Key field: OASVNDCP  
can be indexed by either of the following:
  - SESNPLID {0 to 4094}:  
Key field for OASESNPL.
  - CSESNPLNM:  
Name associated with SESNPLID.
- Info field: OASVNDCP\_INDEX\_REGISTERINFO  
This name can be up to 16 characters long.

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with functional group OASVNDCP.

Functional group ENSV Enhanced Services (ENSV0001) introduces OM group OASVNDCP through the Operator Services AIN (ENSV0014) functionality.

## Registers

The following table lists the registers associated with OM group OAASVNDGP and what they measure. For a description of a register, click on the register name.

### Registers for OM group OASVNDGP

| Register name            | Measures                  |
|--------------------------|---------------------------|
| <a href="#">SBTIMOUT</a> | Session Begin Time Out    |
| <a href="#">NDCALERR</a> | Call Error                |
| <a href="#">NDMSGICL</a> | Message Invalid Call      |
| <a href="#">OSCCLERR</a> | OSAC Call Error           |
| <a href="#">OSCMICL</a>  | OSAC Message Invalid Call |

#### SBTIMOUT

**Register type**

Peg

**Description**

Session Begin Time Out

Pegged when a Session Begin timer expires.

**Note:** For test case(s), datafill OAFUNDEF with a Session Begin timer for a function. Route a call to that function, but do not respond to the Session Begin sent to the simulator. Verify the new OM is pegged after the timer period elapses.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

#### NDCALERR

**Register type**

Peg

**Description**

## Call Error

Pegged when a node requests to end a call due to an unrecoverable error.

**Note:** For test case(s), start by making an OSSAIN call that routes to a service node. Once at the node, perform an "End Call" and send an Abort Call datablock with the "call handling" field set to "Error Recovery."

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**NDMSGICL****Register type**

Peg

**Description**

Message Invalid Call

Pegged when a node attempts to send a message for an invalid call (one that the switch no longer considers active).

**Note:** For test case(s), start by making an OSSAIN call that routes to a service node. Once at the node, make a call that routes to a service node, capture the callId, then end the call. Make a subsequent request from the node which uses the previous callId.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**OSCCLERR****Register type**

Peg

**Description**

OSAC Call Error

This register is pegged when the OSAC Host has to end the call due to an error at the Host.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

OSAC 600

**OSCMICL****Register type**

Peg

**Description**

OSAC Message Invalid Call

This register is pegged when the OSAC Remote receives a message from the Host for a call that is no longer active.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



## OFF250

### Description

OM group Offhook 250 (OFF250) was created for the Offhook Queuing (OHQ) feature.

The following table lists the key and info fields associated with OM group OFF250:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group OFF250.

### Registers

The following table lists the registers associated with OM group OFF250 and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OFF250

| Register name            | Measures              |
|--------------------------|-----------------------|
| <a href="#">OHQOFFD</a>  | OHQ offered           |
| <a href="#">OHQABAND</a> | OHQ abandons          |
| <a href="#">OHQCCBLM</a> | OHQ call limit        |
| <a href="#">OHQTRKLM</a> | OHQ trunk group limit |

#### OHQOFFD

##### Register type

Peg

##### Description

OHQOFFD increments the first time a call is offered OHQ.

##### Associated registers

None

##### Extension registers

OHQOFFD2

**Associated logs**

None

**OHQABAND****Register type**

Peg

**Description**

OHQABAND increments whenever a caller offered OHQ abandons the call.

**Associated registers**

None

**Extension registers**

OHQABAN2

**Associated logs**

None

**OHQCCBLM****Register type**

Peg

**Description**

OHQCCBLM increments every time a call is treated because the maximum number of calls in the OHQ limit is reached.

**Associated registers**

None

**Extension registers**

OHQCCBL2

**Associated logs**

None

**OHQTRKLM****Register type**

Peg

**Description**

OHQTRKLM increments every time a call is treated because the maximum number of calls queued for a trunk group limit is reached.

**Associated registers**

None

**Extension registers**

OHQTRKL2

**Associated logs**

None

## OFZ

### Description

OM group Office Traffic Summary (OFZ) provides information for traffic analysis using primary route scoring. OFZ counts calls for the intended destination, rather than the destination where the call terminates. OFZ records the structure of traffic that arrives at an office, the first routing, and the routing of outgoing traffic.

The system routes a call to a tone or announcement if:

- the tone or announcement is the intended destination of the call
- an error condition occurs that includes a tone or announcement as a part of its treatment

If the treatment routes the call to another tone or announcement, OFZ counts only the first tone or announcement.

The following table lists the key and info fields associated with OM group OFZ. OFZ provides one tuple for each office.

| Key field | Info field |
|-----------|------------|
| None      | None       |

The value of OFFICETYPE in table OFCSTD controls the output of OFZ. All the registers are output in OFFICETYPEs OFF100, OFFCOMB, OFFCOMBLWW, OFFCOMBTOPS, OFF250IBN, OFF100OESD, or OFFCOMBOESD.

The following registers are output in OFFICETYPEs OFF200, OFF200TOPS, OFF200300, OFF250, OFF300, or OFF200OESD: [INANN](#), [INLKT](#), [INOUT](#), [INTONE](#), [NIN](#), [OUTNWAT](#), [OUTMFL](#), [OUTRMFL](#), [OUTOSF](#), [OUTROSE](#), [INABNM](#), [INABNC](#).

### Related functional groups

The following functional groups are related to OM group OFZ:

- DMS-100 Local
- DMS-100/200 Combined Local and Toll
- DMS-100 Wireless Combined Local and Toll with Wireless
- DMS-100/200 Combined Local and Toll with TOPS
- DMS-200 Toll

- DMS-200 Toll with TOPS
- DMS-200/300 Combined Toll and Gateway
- DMS-300 Gateway
- DMS-250 Tandem
- DMS250/SL-100 Combined Tandem and SL-100
- DMS-100 Austrian Local
- DMS-200 Austrian Toll
- DMS-100/200 Austrian Combined Local and Toll

## Registers

The following table lists the registers associated with OM group OFZ and what they measure. For a description of a register, click on the register name.

**Note:** OM group OFZ provides one tuple for each office.

### Registers for OM group OFZ (Sheet 1 of 2)

| Register name            | Measures                                 |
|--------------------------|------------------------------------------|
| <a href="#">INABNC</a>   | Incoming calls abandoned by the customer |
| <a href="#">INABNM</a>   | Incoming calls abandoned by the machine  |
| <a href="#">INANN</a>    | Incoming call to an announcement         |
| <a href="#">INLKT</a>    | Incoming calls to lockout                |
| <a href="#">INOUT</a>    | Incoming to outgoing                     |
| <a href="#">INTONE</a>   | Incoming call to tone                    |
| <a href="#">INTRM</a>    | Incoming to terminating                  |
| <a href="#">LNMBPC</a>   | Line manual busy peg count               |
| <a href="#">NIN</a>      | Number of incoming calls                 |
| <a href="#">NORIG</a>    | Number of originating calls              |
| <a href="#">ORIGABDN</a> | Originating calls abandoned              |
| <a href="#">ORIGANN</a>  | Originating call to announcement         |
| <a href="#">ORIGLKT</a>  | Originating call to lock-out             |

**Registers for OM group OFZ (Sheet 2 of 2)**

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">ORIGOUT</a>  | Originating to outgoing          |
| <a href="#">ORIGTONE</a> | Originating call to tone         |
| <a href="#">ORIGTRM</a>  | Originating to terminating       |
| <a href="#">OUTMFL</a>   | Outgoing match failures          |
| <a href="#">OUTNWAT</a>  | Outgoing network attempts        |
| <a href="#">OUTOSF</a>   | Outgoing original seize failures |
| <a href="#">OUTRMFL</a>  | Outgoing retrial match failures  |
| <a href="#">OUTROSF</a>  | Outgoing retrial seize failures  |
| <a href="#">TRMBLK</a>   | Terminating blocks               |
| <a href="#">TRMMFL</a>   | Terminating match failures       |
| <a href="#">TRMNWAT</a>  | Terminating network attempts     |

**INABNC****Register type**

Peg

**Description**

INABNC counts incoming calls the subscriber abandons before processing. The calls do not require treatment.

**Associated registers**

The following registers are associated with INABNC:

- [INABNM](#)
- OFZ2NET1\_ICABNC
- TRK\_PRERTEAB counts incoming calls that the machine or subscriber abandons. The trunk group counts the calls.
- OTS\_INCABNC counts incoming calls the subscriber abandons before the connection.

**Validation formulas**

The following formulas relate to INABNC and its associated registers:

- $\Sigma \text{TRK\_PRERTEAB} = \text{OFZ\_INABNM} + \text{OFZ\_INABNC}$

**Note:** This relationship does not apply to calls that originate from a mobile telephone exchange (MTX).

- $\text{OFZ\_INABNC} = \text{OTS\_INCABNC}$

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK162

**INABNM****Register type**

Peg

**Description**

INABNM counts incoming calls the machine abandons before processing. The machine abandons calls when they time out at the upstream office while waiting for a receiver, and when an equipment problem exists.

**Associated registers**

The following registers are associated with INABNM:

- [INABNC](#)
- OFZ2NET1\_ICABNM
- TRK\_PRERTEAB counts incoming calls that the machine or subscriber abandons. The trunk group counts the calls.
- OTS\_INCABNCM counts incoming calls the machine abandons before the connection.

**Validation formulas**

The following formulas relate to OFZ and its associated registers:

- $\Sigma \text{TRK\_PRERTEAB} = \text{OFZ\_INABNM} + \text{OFZ\_INABNC}$

**Note:** This relationship does not apply to calls that originate from a mobile telephone exchange (MTX).

- $\text{OFZ\_INABNM} = \text{OTS\_INCABNM}$

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK162

**INANN****Register type**

Peg

**Description**

INANN counts incoming calls that the system routes to a treatment that sends the call to an announcement. The announcement is either the result of a treatment applied during inpulsing, or the intended result of the call. INANN counts the call before it attempts to get a network connection. The register counts the calls once.

**Associated registers**

The following registers are associated with INANN:

- [ORIGANN](#)
- OFZ2NET1\_ICANN
- ANN\_ANNATT counts attempts to generate announcements

**Validation formula**

$\Sigma \text{ ANN\_ANNATT OFZ\_INANN} + \text{OFZ\_ORIGANN}$

**Extension registers**

None

**Associated logs**

TRK138

**INLKT****Register type**

Peg

**Description**

INLKT counts incoming failed calls that the system routes to lockout. The calls fail for one of the following reasons:

- the incoming trunk loses its true identity
- the system cannot connect the call to a tone or announcement



- a forced release initiates manually
- a forced release initiates because call processing requests a delay (CP\_WAITDENY counts the call)

**Associated registers**

The following registers are associated with INLKT:

- OFZ2NET1\_ICLKT
- OTS\_INCLKT counts incoming calls that fail to connect or receive treatment that routes the calls to lockout.

**Validation formula**

$OFZ\_INLKT = OTS\_INCLKT - (\text{number of calls that fail because of remote-end lockout})$

**Extension registers**

None

**Associated logs**

TRK111, TRK113, TRK122, TRK123

**INOUT****Register type**

Peg

**Description**

INOUT counts incoming calls from:

- trunks
- preset conferences
- originating test lines
- auxiliary operator services system (AOSS) positions
- terminating ARTER trunk test facilities that the system routes at the start to trunks, TOPS, or AOSS positions

INOUT also counts TOPS calls that operate coin stations over trunks that use the line number method.

**Associated registers**

The following registers are associated with INOUT:

- OFZ2NET1\_NICOG
- TRK\_TANDEM counts trunk-to-trunk calls, except trunk-to-TOPS calls. The incoming trunk group counts the calls.

**Validation formula**

$\Sigma \text{TRK\_TANDEM} + \text{Trunk-to-TOPS calls} = \text{OFZ\_INOUT} + (\text{OFZ\_INOUT2} \times 65536)$

**Extension registers**

INOUT2

**Associated logs**

None

**INTONE****Register type**

Peg

**Description**

INTONE counts incoming calls that the system routes to a tone. The tone is the result of a treatment applied impulsing, or the tone is the intended result of the call. The register counts the call before it attempts to find a network connection. INTONE counts a call that the system routes to a tone once.

**Associated registers**

The following registers are associated with INTONE:

- [ORIGTONE](#)
- OFZ2NET1\_ICTONE
- TONES\_TONEATT counts attempts to attach a call to a tone.

**Validation formula**

$\Sigma \text{TONES\_TONEATT OFZ\_INTONE} + \text{OFZ\_ORIGTONE}$

**Extension registers**

None

**Associated logs**

TRK138

**INTRM****Register type**

Peg

**Description**

INTRM counts incoming calls that the system routes to a line.

**Associated registers**

OFZ2NET1\_ICTRM

**Extension registers**

INTRM2

**Associated logs**

None

**LNMBPC****Register type**

Peg

**Description**

LNMBPC counts manual busy lines. POTS lines are pegged by one when they become manual busy. Pegging of a P-phone/PSET/DATA/ISDN line depends on the number of virtual identifiers (VIDs) associated with the line when it becomes manual busy, either by LTP/BSY or maintenance action. The line can be pegged more than once.

**Associated registers**

OFZ2NET1\_LNMBSYPC

**Extension registers**

None

**Associated logs**

None

**NIN****Register type**

Peg

**Description**

NIN counts incoming calls that the central control recognizes. The intended destination of the call is a line, trunk, announcement, or tone. NIN counts calls after a call control block and a call process are obtained. The register counts the calls before impulsing is set up.

**Associated registers**

The following registers are associated with NIN:

- [INABNC](#), [INANN](#), [INLKT](#), [INOUT](#), [INTRM](#), [INABNM](#), [INTONE](#)
- OFZ2NET1\_LNMBSYPC
- TRK\_INCATOT and OTS\_NINC count incoming calls. TRK counts calls by trunk group.

**Validation formulas**

The following formulas relate to NIN and its associated registers:

- $OFZ\_NIN + (OFZ\_NIN2 \times 65536) = \Sigma TRK\_INCATOT$
- $OFZ\_NIN + (OFZ\_NIN2 \times 65536) = OTS\_NINC + (OTS\_NINC2 \times 65536)$

**Extension registers**

NIN2

**Associated logs**

None

**NORIG****Register type**

Peg

**Description**

NORIG counts originating calls recognized by the central control. After a call condense block and a call process are obtained. NORIG counts the calls before dialing is set up. NORIG can count a single call at least once. The call is a single call only from the caller point of view. The system counts a three-way call when the flashing switch hook recognizes a correct feature origination signal. The feature origination signal is for the flashing line.

**Associated registers**

The following registers are associated with NORIG:

- [ORIGABDN](#), [ORIGANN](#), [ORIGLKT](#), [ORIGOUT](#), [ORIGTONE](#), [ORIGTRM](#)
- OFZ2NET1\_LINEUSG
- LMD\_NORIGATT and OTS\_NORG count originating calls. LMD counts calls by line module.

**Validation formula**

$OFZ\_NORIG = \Sigma LMD\_NORIGATT = OTS\_NORG$

**Extension registers**

NORIG2

**Associated logs**

None

**ORIGABDN****Register type**

Peg

**Description**

ORIGABDN counts originating calls that the system abandons before the system routes the calls to a trunk, line, or treatment.

**Associated registers**

The following registers are associated with ORIGABDN:

- OFZ2NET1\_LINEABDN
- LMD\_ORIGABN and OTS\_ORGABDN count originating calls that the system abandons before it routes the calls to a trunk, line, or treatment. LMD counts calls that the system does not route through an extended multiprocessor system (XMS)-based peripheral module (XPM).

**Validation formula**

$OFZ\_ORIGABDN = S \text{ } LMD\_ORIGABN = OTS\_ORGABDN$

**Note:** The formula is true only if OFZ\_ORIGABDN and OTS\_ORGABDN are pegged at the same time.

**Extension registers**

None

**Associated logs**

LINE106, LINE108

**ORIGANN****Register type**

Peg

**Description**

ORIGANN counts originating calls that the system routes to an announcement. The announcement can be the result of a treatment during inpulsing, or the intended result of the call. The system counts the call in ORIGANN before an attempt to find a network connection occurs. In GL04, a call is not counted in ORIGANN again if it has been counted in ORIGANN or ORIGINONE.

**Associated registers**

The following registers are associated with ORIGANN:

- [INANN](#)
- OFZ2NET1\_LINEANN
- ANN\_ANNATT counts attempts to attach to announcements.

**Validation formula**

$\Sigma \text{ ANN\_ANNATT OFZ\_INANN} + \text{OFZ\_ORIGANN}$

**Extension registers**

None

**Associated logs**

LINE138

**ORIGLKT****Register type**

Peg

**Description**

ORIGLKT counts originating calls that fail on the destination that the system routes to lock out. The register counts the calls when they do not connect and when the system does not route the call to a treatment. The call fails for one of the following reasons:

- line load control (line is dead)
- a speech link is not available. (The call is queued until a speech link becomes available. If the caller remains off-hook, the call can be successful but ORIGLKT only increases once.)
- a Digitone receiver, or a network connection to a Digitone receiver is not available. (If the caller remains off-hook, the call clears when the problem is cleared but ORIGLKT increases once.)

**Note:** In GL04, ORIGLKT will be incremented when a lockout maintenance instruction is performed on an originating call as a result of a treatment. This OM will not be incremented if either ORIGANN or ORIGTONE has already been incremented.

**Associated registers**

The following registers are associated with ORIGLKT:

- OFZ2NET1\_LINELKT
- The system counts OTS\_ORGLKT originating calls that fail and that the system routes to lockout. The system counts the calls when they do not connect and route to a treatment.

**Validation formula**

OFZ\_ORIGLKT = OTS\_ORGLKT

**Extension registers**

None

**Associated logs**

LINE104, LINE105, LINE109, LINE204, NET130, OM2200

**ORIGOUT****Register type**

Peg

**Description**

ORIGOUT counts originating calls that the system routes to a trunk or a test facility.

**Associated registers**

OFZ2NET1\_LINEOG

**Extension registers**

ORIGOUT2

**Associated logs**

None

**ORIGTONE****Register type**

Peg

**Description**

ORIGTONE counts originating calls that route to a tone. The register counts a call before it attempts to find a network connection. The tone is either the result of a treatment or the intended result of the call. ORIGTONE counts calls that the system routes to a treatment that in turn routes the call to a tone. ORIGTONE counts the call once.

Beginning in GL04, a call is not counted in register ORIGANN again if it has been counted in register ORIGANN or ORIGTONE.

**Associated registers**

The following registers are associated with ORIGIN\_TONE:

- [INTONE](#)
- OFZ2NET1\_LINETONE
- TONES\_TONEATT counts attempts to attach to tones.

**Validation formula**

$\Sigma (\text{TONES\_TONEATT}) \text{ OFZ\_INTONE} + \text{OFZ\_ORIGIN\_TONE}$

**Extension registers**

None

**Associated logs**

LINE138

**ORIGIN\_TONE****Register type**

Peg

**Description**

ORIGIN\_TONE counts originating calls that the system routes to a line. The register counts the call if a line is available or is not available.

**Associated registers**

OFZ2NET1\_LINETONE

**Extension registers**

ORIGIN\_TONE2

**Associated logs**

None

**OUTMFL****Register type**

Peg

**Description**

OUTMFL counts calls that fail to find a network path to a selected outgoing or test trunk on the first attempt. A second attempt occurs to find an idle trunk and a network path.



**Associated registers**

The following registers are associated with OUTMFL:

- OFZ2NET1\_OGMFL
- OUTMFL and SOTS\_SOUTMFL count first trial match failures. SOTS\_SOUTMFL counts calls that fail to find a network path from a line and that trunk to a selected outgoing or test trunk.
- TRK\_OUTMTCHF counts match failures by trunk group.

**Validation formulas**

The following formulas relate to OUTMFL and its associated registers:

- $OFZ\_OUTMFL = SOTS\_SOUTMFL$
- $\Sigma TRK\_OUTMTCHF = OFZ\_OUTMFL + OFZ\_OUTRMFL$

**Extension registers**

None

**Associated logs**

NET130

**OUTNWAT****Register type**

Peg

**Description**

OUTNWAT counts incoming and originating calls that are intended for an exact outgoing or test trunk. A single call can use two or more network paths to different ports of the service circuit. For example, connection by a conference circuit or digital echo suppressor requires more than one network path.

**Associated registers**

The following registers are associated with OUTNWAT:

- OFZ2NET1\_OGNWAT
- After OUTNWAT counts the call, one of the following actions occurs:
  - TRK\_CONNECT counts a call after it connects.
  - After a first trial failure, the system routes the call in an attempt to select another outgoing trunk. [OUTMFL](#) and TRK\_OUTMTCHF count the call.
  - After failure to get path followed by network blockage heavy traffic (NBLH) treatment, [OUTMFL](#) and TRK\_OUTMTCHF count the call.

- After failure to get a path followed by no treatment, TRK\_OUTFAIL counts the call.
- If double seizure of a trunk occurs, TRK\_GLARE counts the call. The system makes a new path selection. If the system again encounters double seizure of a trunk, the call routes to a generalized no-circuit (GNCT) treatment.
- SOTS\_SOUTNWT counts the attempts to find a network path from a line or trunk to a selected outgoing or test trunk.

**Validation formulas**

The following formulas relate to OUTNWAT and its associated registers:

- $OFZ\_OUTNWAT + (OFZ\_OUTNWAT2 \times 65536) = OFZ\_OUTMFL + OFZ\_OUTRMFL + \Sigma (TRK\_CONNECT + TRK\_GLARE + TRK\_OUTFAIL + TRK\_OUTMTCHF)$
- $OFZ\_OUTNWAT + (OFZ\_OUTNWAT2 \times 65536) = SOTS\_SOUTNWT + (SOTS\_SOUTNWT2 \times 65536)$

**Extension registers**

OUTNWAT2

**Associated logs**

None

**OUTOSF****Register type**

Peg

**Description**

OUTOSF counts calls that fail to seize an outgoing trunk on the first attempt after network paths are acquired. A second attempt occurs to find an idle trunk and a network path, and to seize the trunk.

One of the following conditions causes a failure:

- a reversed trunk
- failure to receive a known start-dial
- not planned stop-dial
- timeout before expected stop-dial
- CCS7 errors

**Associated registers**

The following registers are associated with OUTOSF:

- OFZ2NET1\_OGOSF
- SOTS\_SOUTOSF counts first trial seize failures that occur after an outgoing trunk is selected and the necessary network paths acquired.

**Validation formula**

$OFZ\_OUTOSF = SOTS\_SOUTOSF$

**Extension registers**

None

**Associated logs**

TRK113, TRK121, TRK162, C7UP111

**OUTRMFL****Register type**

Peg

**Description**

OUTRMFL counts calls that fail on the second attempt to find a network path to a selected outgoing or test trunk. This register is not incremented in GL04.

**Associated registers**

The following registers are associated with OUTRMFL:

- [OUTMFL](#)
- OFZ2NET1\_OGTRMFL
- OUTRMFL and SOTS\_SOUTRMFL count second trial match failures.
- The system counts TRK\_OUTMTCHF match failures. The trunk group counts failures.

**Validation formulas**

The following formulas relate to OUTRMFL and its associated registers:

- $\Sigma TRK\_OUTMTCHF = OFZ\_OUTMFL + OFZ\_OUTRMFL$
- $OFZ\_OUTRMFL = SOTS\_SOUTRMFL$

**Extension registers**

None

**Associated logs**

None

**OUTROSF****Register type**

Peg

**Description**

OUTROSF counts calls that fail on the second attempt to seize an outgoing trunk. This attempt occurs after the network paths are acquired. One of the following conditions can cause a failure:

- a reversed trunk
- failure to receive a known start-dial
- unplanned stop-dial
- time-out before an expected stop-dial

The system disconnects the call after the second failure and the call receives start signal timeout (SSTO) treatment. An equal access call receives signal timeout BOC (STOB) or signal timeout IC/INC (STOC) treatment.

The system increases OUTROSF when a second attempt occurs to run a continuity test (COT) for an outgoing ISUP trunk. The second attempt occurs if the first COT attempt fails.

**Associated registers**

The following registers are associated with OUTROSF:

- OFZ2NET1\_OGROSF
- SOTS\_SOUTROSF counts calls that fail the second attempt to seize an outgoing trunk

**Validation formula**
$$\text{OFZ\_OUTROSF} = \text{SOTS\_SOUTROSF}$$
**Extension registers**

None

**Associated logs**

TRK113, TRK121, TRK162

**TRMBLK****Register type**

Peg

**Description**

TRMBLK counts attempts to obtain a voice path to a terminating line that fails. This failure occurs when no free channel is present between the host network and the terminating line.

The system counts more than one failed attempt if part of a hunt group directs the call. The system also counts each attempt in OFZ registers TRMMFL and TRMNBWAT. The terminating line control device also counts in LMD registers NTERMATT and TERRMBLK. If no alternate line is available, the system routes the call to network blockage normal traffic (NBLN) treatment. Register TRMTRS\_TRSNBLN counts the calls.

**Associated registers**

The following registers are associated with TRMBLK:

- OFZ2NET1\_TRMBLK
- LMD\_TERMBLK counts failures in the line-to-network segment. The register counts call failures for non-XPM modules.
- SOTS\_STRMBLK counts attempts to find a voice path from the network to a terminating line that fails. Failures occur when:
  - all the LM channels to the network are busy
  - the idle channels on lines to the network and line shelves that serve the terminating line are not linked

**Validation formulas**

The following formulas relate to TRMBLK and its associated registers:

- $OFZ\_TRMBLK = S \text{ LMD\_TERMBLK}$
- $OFZ\_TRMBLK = SOTS\_STRMBLK$

**Extension registers**

None

**Associated logs**

NET130, TRK138, LINE138

**TRMMFL****Register type**

Peg

**Description**

TRMMFL counts failed attempts to find a voice path to a terminating line. The system counts more than one failed attempt if the call goes to a part of a hunt group. The system counts each attempt in OFZ register

TRMNWAT and in LMD\_NTERMATT for the terminating line control device.

Failure in the path search sequence can occur if the host switch network cannot obtain a path (that is, a free channel on a link between the host switch network and the terminating line). If the network cannot find an alternate path, the system routes the call to network blockage heavy traffic (NBLH) treatment. Registers TRMBLK and TERMBLK count the failure to obtain a free channel.

If no alternate line is available, the system routes the call to network blockage normal traffic (NBLN) treatment. Register TRMTRS\_TRSNBLN counts the calls.

### **Associated registers**

The following registers are associated with TRMMFL:

- OFZ2NET1\_TRMMCHFL
- SOTS\_STRMMFL counts attempts to find a voice path to a terminating line that fail because a network connection is not available.

### **Validation formula**

OFZ\_TRMMFL = SOTS\_STRMMFL

### **Extension registers**

None

### **Associated logs**

NET130, LINE138, TRK138

## **TRMNWAT**

### **Register type**

Peg

### **Description**

TRMNWAT counts attempts to find a voice path to a terminating line. The complete path includes the following elements:

- a segment through the network
- a channel on the link between the line module and the network
- a matching channel on the line shelf

TRMNWAT counts a call only for each attempt. The calls count each attempt, whether it succeeds or fails.

**Associated registers**

The following registers are associated with TRMNWAT:

- OFZ2NET1\_TRMLNAT
- LMD\_NTERMATT counts intra-office calls for each line module.
- SOTS\_STRMNWT counts attempts to find a voice path to a terminating line.

**Validation formulas**

The following formulas relate to TRMNWAT and its associated registers:

- $OFZ\_TRMNWAT + (OFZ\_TRMNWAT2 \times 65536) = \Sigma LMD\_NTERMATT$
- $OFZ\_TRMNWAT + (OFZ\_TRMNWAT2 \times 65536) = SOTS\_STRMNWAT + (SOTS\_STRMNWAT2 \times 65536)$

**Extension registers**

TRMNWAT2

**Associated logs**

None

## OFZ2

### Description

OM group Office Traffic Extension Summary (OFZ2) counts calls that the system routes to generalized no circuit treatment (GNCT). The system routes a call to GNCT when a trunk group is the last route in the route list and all trunks are busy.

The OM group has 13 registers. These registers give the cause of the GNCT for outgoing trunks or for the outgoing side of two-way trunks. The name of each register corresponds to an entry in the no circuit class field, NCCLS in table TRKGRP.

The following table lists the key and info fields associated with OM group OFZ2:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There following functional groups are associated with OM group OFZ2:

- OFF100 Local
- OFFCOMB Combined local/toll
- OFFCOMBTOPS Combined local/toll with TOPS
- OFF200 Toll
- OFF200TOPS Toll with TOPS
- OFF200300 Combined gateway/toll
- OFF300 Gateway
- OFF250 DMS-250
- OFF250IBN DMS-250/SL-100
- OFF100OESD Austrian local
- OFF200OESD Austrian toll
- OFFCOMBOESD Austrian combined local/toll



## Registers

The following table lists the registers associated with OM group OFZ2 and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OFZ2

| Register name           | Measures                           |
|-------------------------|------------------------------------|
| <a href="#">OFZNCBN</a> | No circuit business network trunks |
| <a href="#">OFZNCID</a> | No circuit inward dial trunks      |
| <a href="#">OFZNCIM</a> | No circuit intermachine trunks     |
| <a href="#">OFZNCIT</a> | No circuit intertoll trunks        |
| <a href="#">OFZNCLT</a> | No circuit local tandem trunks     |
| <a href="#">OFZNCOF</a> | No circuit offnet trunks           |
| <a href="#">OFZNCON</a> | No circuit connect trunks          |
| <a href="#">OFZNCOT</a> | No circuit other trunk             |
| <a href="#">OFZNCRT</a> | No circuit trunks                  |
| <a href="#">OFZNCTC</a> | No circuit toll completing trunks  |
| <a href="#">OFZNOSC</a> | No service circuit trunks          |
| <a href="#">PDLM</a>    | Machine dialed partial dials       |
| <a href="#">PSGM</a>    | Machine dialed permanent signal    |

### OFZNCBN

#### Register type

Peg

#### Description

OFZNCBN counts calls the system routes to generalized no circuit treatment (GNCT). The system routes the calls to GNCT because Meridian Digital Centrex (MDC) trunk is not available.

#### Associated registers

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCID****Register type**

Peg

**Description**

OFZNCID counts calls the system routes to generalized no circuit treatment (GNCT). The system routes a call to GNCT when a direct inward dial or direct outward dial trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCIM****Register type**

Peg

**Description**

OFZNCIM counts calls that the system routes to generalized no circuit treatment (GNCT). The system routes the calls to GNCT because a circuit intermachine trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCIT****Register type**

Peg

**Description**

OFZNCIT counts calls that the system routes to generalized no circuit treatment (GNCT) because an intertoll trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCLT****Register type**

Peg

**Description**

OFZNCLT counts calls that the system routes to generalized no circuit treatment (GNCT) because a local tandem trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCOF****Register type**

Peg

**Description**

OFZNCOF counts calls that the system routes to generalized no circuit treatment (GNCT). The system routes the calls to GNCT because circuit offnet access or direct dial trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCON****Register type**

Peg

**Description**

OFZNCON counts calls that the system routes to generalized no circuit treatment (GNCT). The system routes the calls to GNCT because dedicated access or mobile telephone exchange trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCOT****Register type**

Peg

**Description**

OFZNCOT counts calls that the system routes to generalized no circuit treatment (GNCT). The system routes the calls to GNCT because one of the following types of trunk is not available:

- test line
- test desk
- maintenance trunks
- AV101

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCRT****Register type**

Peg

**Description**

OFZNCRT counts calls that the system routes to GNCT. The system routes a call to GNCT because one of the following types of trunk is not available:

- 0+/0- tandem to TOPS
- outgoing to AMR2 or CAMA
- outgoing local
- recording completing outgoing
- TOPS outgoing

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCCTC, OFZNCCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCN, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNCTC****Register type**

Peg

**Description**

OFZNCTC counts calls that the system routes to generalized no circuit treatment (GNCT) because toll completing trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**OFZNOSC****Register type**

Peg

**Description**

OFZNOSC counts calls the system routes to generalized no circuit treatment (GNCT). The system routes the call because automatic number announcement or automatic intercept trunk is not available.

**Associated registers**

TRMTRS\_TRSGNCT, which counts calls that the system routes to GNCT.

**Validation formula**

Register TRMTRS\_TRSGNCT = The sum of the OFZ2 Registers, OFZNCIT, OFZNCTC, OFZNCLT, OFZNCBN, OFZNCID, OFZNOSC, OFZNCOT, OFZNCRT, OFZNCIM, OFZNCON, OFZNCOF

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**PDLM****Register type**

Peg

**Description**

PDLM counts the machine-dialed calls that the system routes to partial dial treatment.

**Associated registers**

TRMTCM\_TCM PDIL, which counts calls that the system routes to partial dial timeout treatment.

**Extension registers**

None

**Associated logs**

None

**PSGM****Register type**

Peg

**Description**

PSGM counts machine-dialed calls that the system routes to permanent signal treatment.

**Associated registers**

TRMTCM\_TCMPSIG, which counts calls that the system routes to permanent signal timeout treatment.

**Extension registers**

None

**Associated logs**

TRK115, TRK117, TRK138, TRK183



## OFZ2NET1

### Description

OM group Office to Network Group 1 (OFZ2NET1) provides traffic analysis information for each bearer network. Using primary route scoring, the group counts calls for the intended destination rather than for the terminating destination. OFZ2NET2 contains registers that are extensions of a subset of registers in OFZ2NET1.

OFZ2NET1 records the structure of traffic that arrives at an office on a particular bearer network, the first routing, and the routing of outgoing traffic. A call routes to a tone or announcement if:

- the tone or announcement is the intended destination of the call.
- an error condition occurs that includes a tone or announcement as a part of the call treatment.

**Note:** The a treatment routes the call to another tone or announcement, OFZ2NET1 counts only the first tone or announcement.

OM groups OFZ and OFZ2 provide traffic analysis information office wide. OFZ2NET1 and OFZ2NET2 provide similar traffic analyses with further refinement on a per-bearer network basis for an office.

New office parameter MULTINET\_DISPLAY\_ACTIVE must be set to Y before the system can display OFZ2NET1.

The following table lists the key and info fields associated with OM group OFZ2NET1. The group provides one tuple for each bearer network datafilled in Table BEARNETS. The group structure depends on the office type of the call server.

| Key field         | Info field    |
|-------------------|---------------|
| Bearer network ID | BEARNETS_INFO |

Info field BEARNETS\_INFO includes:

- the bearer network ID (NET 0)
- the bearer network name (TDM\_ENET)

The following table describes the relationship between incoming calls and registers in OFZ2NET1. For a complete list of registers in OFZ2NET1, refer to the section [Registers](#).

| Register name           | Intended destination      | Routing          |
|-------------------------|---------------------------|------------------|
| <a href="#">NICO</a>    | trunk                     | trunk            |
| <a href="#">ICTRM</a>   | line                      | line             |
| <a href="#">ICANN</a>   | trunk, line, announcement | announcement     |
| <a href="#">ICTONE</a>  | trunk, line, tone         | tone             |
| <a href="#">ICLKT</a>   | trunk, line               | lockout          |
| <a href="#">ICABNC</a>  | trunk, line               | customer abandon |
| <a href="#">ICABNM</a>  | trunk, line               | machine abandon  |
| <a href="#">TRKORIG</a> | all                       | all              |

The following table describes the relationship between originating calls and registers in OFZ2NET1. For a complete list of registers in OFZ2NET1, refer to the section [Registers](#).

| Register name            | Intended destination      | Routing      |
|--------------------------|---------------------------|--------------|
| <a href="#">LINEOG</a>   | trunk                     | trunk        |
| <a href="#">LINETRM</a>  | line                      | line         |
| <a href="#">LINEANN</a>  | trunk, line, announcement | announcement |
| <a href="#">LINETONE</a> | trunk, line, tone         | tone         |
| <a href="#">LINELKT</a>  | trunk, line               | lockout      |
| <a href="#">LINEABDN</a> | trunk, line               | abandon      |
| <a href="#">LINEUSG</a>  | trunk, line               | all          |

The following table describes the relationship between terminating calls and registers in OFZ2NET1. For a complete list of registers in OFZ2NET1, refer to the section [Registers](#).

| Register name            | Event                                                                              |
|--------------------------|------------------------------------------------------------------------------------|
| <a href="#">TRMMCHFL</a> | Network blockage heavy traffic (NBLH) or<br>Network blockage normal traffic (NBLN) |
| <a href="#">TRMBLCK</a>  | Network blockage normal traffic (NBLN)                                             |
| <a href="#">LNMBSYPC</a> | Line manual busy                                                                   |
| <a href="#">TRMLNAT</a>  | All outgoing traffic                                                               |

Office parameter OFFICETYPE in table OFCSTD controls the generation of the registers for OFZ2NET1. The following table describes the correct entries (values) for table OFCSTD. When the system reports an answer indicator to CM from the outgoing trunk, the OM Answer register for the outgoing trunk increases.

#### Office type values for table OFCSTD (Sheet 1 of 2)

| Value (name) | Office type                                              |
|--------------|----------------------------------------------------------|
| NOOFFICE     | Default                                                  |
| OFF100       | Local                                                    |
| OFF100OESD   | Austrian local                                           |
| OFF100SCP    | DMS-100 Service Control Point (SCP)                      |
| OFF200       | Toll                                                     |
| OFF200AVON   | DMS 100/200 local/toll with Autovon<br>(U.S. and Canada) |
| OFF200OESD   | Australian toll                                          |
| OFF200TOPS   | Toll with traffic operator position system<br>(TOPS)     |
| OFF200300    | Combined gateway/toll                                    |
| OFF250       | DMS-250                                                  |
| OFF250IBN    | DMS-250/SL-100                                           |

**Office type values for table OFCSTD (Sheet 2 of 2)**

| Value (name)    | Office type                                                     |
|-----------------|-----------------------------------------------------------------|
| OFF300          | Gateway                                                         |
| OFF500          | DMS-500                                                         |
| OFFCOMB         | Combined local/toll                                             |
| OFFCOMB300      | Combined local/toll and gateway                                 |
| OFFCOMB300ITOPS | Combined local/toll and gateway with international TOPS (ITOPS) |
| OFFCOMBITOPS    | Combined local/toll with ITOPS                                  |
| OFFCOMBLWW      | Combined local/toll with wireless                               |
| OFFCOMBOESD     | Australian combined local/toll                                  |
| OFFCOMBTOPS     | Combined local/toll with TOPS                                   |
| OFFMTX100I      | DMS-MTX with DMS100I                                            |

**Related functional groups**

The following functional groups are associated with OM group OFZNET1:

- Carrier VoIP North America
- Carrier VoIP International

**Registers**

The following table lists the registers associated with OM group OFZ2NET1 and what they measure. For a description of a register, click on the register name.

**Registers for OM group OFZ2NET1 (Sheet 1 of 2)**

| Register name          | Measures                                 |
|------------------------|------------------------------------------|
| <a href="#">ICANN</a>  | Incoming calls to an announcement        |
| <a href="#">ICABNM</a> | Incoming calls abandoned (by machine)    |
| <a href="#">ICABNC</a> | Incoming calls abandoned (by subscriber) |
| <a href="#">ICLKT</a>  | Incoming calls to lockout                |

**Registers for OM group OFZ2NET1 (Sheet 2 of 2)**

| Register name                   | Measures                          |
|---------------------------------|-----------------------------------|
| <a href="#"><u>ICTONE</u></a>   | Incoming calls to tone            |
| <a href="#"><u>ICTRM</u></a>    | Incoming calls to terminating     |
| <a href="#"><u>LINEABDN</u></a> | Originating calls abandoned       |
| <a href="#"><u>LINEANN</u></a>  | Originating calls to announcement |
| <a href="#"><u>LINELKT</u></a>  | Originating calls to lockout      |
| <a href="#"><u>LINEOG</u></a>   | Originating calls to outgoing     |
| <a href="#"><u>LINETONE</u></a> | Originating calls to tone         |
| <a href="#"><u>LINETRM</u></a>  | Originating calls to terminating  |
| <a href="#"><u>LINEUSG</u></a>  | Number of originating calls       |
| <a href="#"><u>LNMBSYPC</u></a> | Line manual busy peg count        |
| <a href="#"><u>NICOG</u></a>    | Incoming to outgoing calls        |
| <a href="#"><u>OGMFL</u></a>    | Outgoing match failures           |
| <a href="#"><u>OGNWAT</u></a>   | Outgoing network attempts         |
| <a href="#"><u>OGOSF</u></a>    | Outgoing original seize failures  |
| <a href="#"><u>OGROSF</u></a>   | Outgoing retrial seize failures   |
| <a href="#"><u>OGTRMFL</u></a>  | Outgoing retrial match failures   |
| <a href="#"><u>TRKORIG</u></a>  | Number of incoming calls          |
| <a href="#"><u>TRMBLCK</u></a>  | Terminating blocks                |
| <a href="#"><u>TRMLNAT</u></a>  | Terminating network attempts      |
| <a href="#"><u>TRMMCHFL</u></a> | Terminating match failures        |

**ICABNC****Register type**

Peg

**Description**

ICABNC counts incoming calls on the associated bearer network that the subscriber abandons before processing. The calls do not require treatment. CABNC is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[ICABNM](#), OFZ\_INABNC

**Extension registers**

None

**Associated Logs**

TRK114, TRK116, TRK162

**ICABNM****Register type**

Peg

**Description**

ICABNM counts incoming calls on the associated bearer network that the machine abandons before processing because:

- the call times out at the upstream office while waiting for a receiver
- an equipment problem exists

ICABNM is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[ICABNC](#), OFZ\_INABNM

**Extension registers**

None

**Associated Logs**

TRK114, TRK116, TRK162

**ICANN****Register type**

Peg

**Description**

ICANN counts incoming calls on the associated bearer network that the system routes to an announcement that is either the result of a treatment applied during origination or the intended result of the call. ICANN counts the call before it attempts to get network connection. The register counts calls that the system routes to a treatment that routes the call to an announcement. ICANN counts the calls once.

ICANN is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[LINEANN](#), OFZ\_INANN

**Extension registers**

None

**Associated Logs**

TRK138

**ICLKT****Register type**

Peg

**Description**

ICLKT counts incoming failed calls that the system routes to lockout. Calls fails because:

- the incoming trunk loses its true identity
- the system cannot connect the call to a tone or announcement
- a forced release initiates manually
- a forced release initiates because call processing requests a delay (CP\_WAITDENY counts the call).

ICLKT is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

OFZ\_INLKT

**Extension registers**

None

**Associated Logs**

TRK111, TRK113, TRK123

**ICTONE****Register type**

Peg

**Description**

ICTONE counts incoming calls on the associated bearer network that route to a tone. The tone is either the result of a treatment applied during origination, or the intended result of the call. ICTONE counts the call before it attempts to find a network connection. The register counts a call once.

ICTONE is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**[LINETONE](#), OFZ\_INTONE**Extension registers**

None

**Associated Logs**

TRK138

**ICTRM****Register type**

Peg

**Description**

ICTRM counts incoming calls on the associated bearer network that route to a line. The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_INTRM

**Extension registers**

ICTRM2



**Associated Logs**

None

**LINEABDN****Register type**

Peg

**Description**

LINEABDN counts originating calls on the associated bearer network that the system abandons before the calls route to a trunk, line or treatment. The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_ORIGABDN

**Extension registers**

None

**Associated Logs**

LINE106, LINE108

**LINEANN****Register type**

Peg

**Description**

LINEANN counts originating calls on the associated bearer network that route to an announcement. The announcement is either the result of a treatment during origination, or the intended result of the call. The register counts a call before an attempt to find a network connection.

LINEANN is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**[ICANN](#), OFZ\_ORIGANN**Extension registers**

None

**Associated Logs**

LINE138

**LINELKT****Register type**

Peg

**Description**

LINELKT counts originating calls on the associated bearer network that fail on the intended destination that route to lockout. The register increments when the calls when either do not connect or do not route to treatment. Calls can fail because:

- line load control (the line is dead)
- a speech link is not available. The call is in queue until a speech link becomes available. If the caller remains off-hook, the call can be successful. LINELKT pegs one time.
- a digitone receiver, or a network connection to a digitone receiver is not available. If the caller remains off-hook, the problem clears when the call is successful. LINELKT pegs one time.

LINELKT is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_ORIGLKT

**Extension registers**

None

**Associated Logs**

LINE104, LINE105, LINE109, LINE204, NET130, OM2200

**LINEOG****Register type**

Peg

**Description**

LINEOG counts originating calls on the associated bearer network that route to a trunk or test facility. The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_ORIGOUT

**Extension registers**

LINEOG2

**Associated Logs**

None

**LINETONE****Register type**

Peg

**Description**

LINETONE counts originating calls on the associated bearer network that route to a tone that is either the result of a treatment, or the intended result of the call. The register counts a call before it attempts to find a network connection. LINETONE counts a call one time.

The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**[ICTONE](#), OFZ\_ORIGTONE**Extension registers**

None

**Associated Logs**

LINE138

**LINETRM****Register type**

Peg

**Description**

LINETRM counts originating calls on the associated bearer network that the system routes to a line. The register counts the call whether a line is available or not. LINETRM is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_ORIGTRM

**Extension registers**

LINETRM2

**Associated Logs**

None

**LINEUSG****Register type**

Peg

**Description**

LINEUSG counts originating calls on the associated bearer network that the core recognizes. The register counts a call after a call condense block (CCB) and a call process are obtained, and before dialing is set up. The call is three-way when the flashing switch hook recognizes a correct feature origination signal for the flashing line.

LINEUSG is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

[LINEABDN](#), [LINEANN](#), [LINELKT](#), [LINEOG](#), [LINETONE](#), [LINETRM](#), OFZ\_NORIG

**Extension registers**

LINEUSG2

**Associated Logs**

None

**LNMBSYPC****Register type**

Peg

**Description**

LNMBSYPC counts lines on the associated bearer network that are manual busy. The register pegs each POTS line by one when the line transitions to a manual busy state. How LNMBSYPC pegs each P-phone/PSET/DATA/ISDN line depends on the number of virtual identifiers (VID) associated with the line when the line becomes manual

busy by maintenance action such as the BSY command at the LTP MAPCI level. The register can peg the P-phone line more than once.

LNMBSYPC is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

OFZ\_LNMBPC

**Extension registers**

None

**Associated Logs**

None

**NICOG****Register type**

Peg

**Description**

NICOG counts incoming calls on the associated bearer network from trunks, preset conferences and originating test lines. The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

OFZ\_INOUT

**Extension registers**

NICOG2

**Associated Logs**

None

**OGMFL****Register type**

Peg

**Description**

OGMFL counts calls on the associated bearer network that fail to find a network path to a selected outgoing or test trunk on the first attempt. A second attempt occurs to find an idle trunk and a network path.

OGMFL is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[OGTRMFL](#), OFZ\_OUTMFL

**Extension registers**

None

**Associated Logs**

NET130

**OGNWAT****Register type**

Peg

**Description**

OGNWAT counts incoming and outgoing calls on the associated bearer network that are intended for a specific outgoing or test trunk. A call can use two or more network paths to different ports for the service circuit. For example, connection by a conference circuit requires more than one network path.

OGNWAT is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

OFZ\_OUTNWAT

**Extension registers**

OGNWAT2

**Associated Logs**

None

**OGOSF****Register type**

Peg

**Description**

OGOSF counts calls on the associated bearer network that fail to seize an outgoing trunk on the first attempt after network paths are acquired.

A second attempt occurs to find an idle trunk and a network path, and to seize the trunk. One of the following conditions causes a failure:

- a reversed trunk
- failure to receive a known start-dial
- an unplanned stop-dial
- a time-out before an expected stop-dial
- CCS7 errors

OGOSF is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[OGROSF](#), OFZ\_OUTOSF

**Extension registers**

None

**Associated Logs**

TRK113, TRK121, TRK162, C7UP111

**OGROSF****Register type**

Peg

**Description**

OGROSF counts calls on the associated bearer network that fail on the second attempt to seize an outgoing trunk. This attempt occurs after the network paths have been acquired. One of the following conditions causes a failure:

- a reversed trunk
- failure to receive a known start-dial
- an unplanned stop-dial
- a time-out before an expected stop-dial
- CCS7 errors

The system disconnects the call after the second failure and the call receives start signal timeout IC/INC (STOC) treatment. An equal access call receives signal.

The system increments OGRSOF when a second attempt occurs to run a continuity test (COT) for an outgoing ISUP trunk. The second attempt occurs if the first COT attempt fails.

The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[OGOSF](#), OFZ\_OUTROSF

**Extension registers**

None

**Associated Logs**

TRK113, TRK121, TRK162, C7UP111

**OGTRMFL****Register type**

Peg

**Description**

OGTRMFL counts calls on the associated bearer network that fail on the second attempt to find a network path to a selected outgoing or test trunk. The register is present in the following offices: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[OGMFL](#), OFZ\_OUTRMFL

**Extension registers**

None

**Associated Logs**

None

**TRKORIG****Register type**

Peg

**Description**

TRKORIG counts incoming calls on the associated bearer network that the core recognizes. The intended destination for these calls can be a



line, trunk, announcement or tone. TRKORIG counts calls after a call control block and call process are obtained. The register can count a call before all incoming digits are received.

TRKORIG is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW, NOOFFICE.

**Associated registers**

[ICABNC](#), [ICABNM](#), [ICANN](#), [ICLKT](#), [NICO](#), [ICTONE](#), [ICTRM](#), OFZ\_NIN

**Extension registers**

TRKORIG2

**Associated Logs**

None

**TRMBLCK****Register type**

Peg

**Description**

TRMBLK counts attempts to obtain a voice path to a terminating line on the associated bearer network that fail because no free channel is present between the host network and the terminating line. The register counts more than one failed attempt if part of a hunt group directs the call. If no alternate line is available, the system routes the call to network blockage normal traffic (NBLN) treatment.

TRMBLK is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

[TRMMCHFL](#), [TRMLNAT](#), OFZ\_TRMBLK

**Extension registers**

None

**Associated Logs**

LINE138, NET 130, TRK138

**TRMLNAT****Register type**

Peg

**Description**

TRMLNAT counts attempts to find a voice path on the associated bearer network to a terminating line. A complete path includes:

- a segment through the network
- a channel on the link between the line module and the network
- a matching channel on the line shelf

TRMLNAT counts a call for each failed or successful attempt. The register is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

[TRMBLCK](#), [TRMMCHFL](#), OFZ\_TRMNLWAT

**Extension registers**

TRMLNAT2

**Associated Logs**

None

**TRMMCHFL****Register type**

Peg

**Description**

TRMMCHFL counts calls on the associated bearer network that fail to find a network path to a terminating line. The register counts more than one failed attempt if the call goes to a part of a hunt group. Failure in the path search sequence occurs when:

- the host switch network cannot obtain a path. The call routes to network blockage heavy traffic (NBLH) treatment
- a free channel on a link between the host switch network and the terminating line cannot be obtained.

If no alternate path is available, the call routes to network blockage normal traffic (NBLN) treatment.

TRMMCHFL is present in the following office types: OFF100, OFFCOMB, OFFCOMBTOPS, OFF100SCP, OFF100OESD, OFFCOMBOESD, OFF250IBN, OFFCOMBITOPS, OFFMTX100I, OFF500, OFFCOMBLWW.

**Associated registers**

[TRMBLCK](#), [TRMLNAT](#), OFZ\_TRMMFL

**Extension registers**

None

**Associated Logs**

LINE138, NET130, TRK138

## OFZ2NET2

### Description

OM group Office to Network Group 2 (OFZ2NET2) provides traffic analysis information for each bearer network. Registers in the group count calls that route to generalized no circuit treatment (GNCT) when a trunk group is the last route in the route list and all trunks are busy.

OFZ2NET2 registers with the prefix *NCKT* count the number of calls by the trunk or call types routed to GNCT for outgoing trunks or for the outgoing side of two-way trunks. Register names correspond to an entry in field No Circuit Class (NCCLS) which is datafilled for each trunk defined in table TRKGRP.

To display OFZ2NET2, set new office parameter MULTINET\_DISPLAY\_ACTIVE to Y.

The following table lists the key and info fields associated with OM group OFZ2NET2. The group provides one tuple for each bearer network datafilled in table BEARNETS. The group structure depends on the office type of the call server.

| Key field         | Info field    |
|-------------------|---------------|
| Bearer network ID | BEARNETS_INFO |

Info field BEARNETS\_INFO includes:

- the bearer network ID (NET 0)
- the bearer network name (TDM\_ENET)

Office parameter OFFICETYPE in table OFCSTD controls the generation of the registers for OFZ2NET2. The following table describes the correct entries (values) for table OFCSTD. When the system reports an answer indicator to CM from the outgoing trunk, the OM Answer register for the outgoing trunk increases. These registers will be displayed for the office types in the following table.

#### Office type values for table OFCSTD (Sheet 1 of 2)

| Value (name) | Office type |
|--------------|-------------|
| NOOFFICE     | Default     |
| OFF100       | Local       |

**Office type values for table OFCSTD (Sheet 2 of 2)**

| <b>Value (name)</b> | <b>Office type</b>                                              |
|---------------------|-----------------------------------------------------------------|
| OFF100OESD          | Austrian local                                                  |
| OFF200              | Toll                                                            |
| OFF200OESD          | Australian toll                                                 |
| OFF200TOPS          | Toll with traffic operator position system (TOPS)               |
| OFF200300           | Combined gateway/toll                                           |
| OFF250              | DMS-250                                                         |
| OFF250IBN           | DMS-250/SL-100                                                  |
| OFF300              | Gateway                                                         |
| OFFCOMB             | Combined local/toll                                             |
| OFFCOMB300ITOPS     | Combined local/toll and gateway with international TOPS (ITOPS) |
| OFFCOMBOESD         | Australian combined local/toll                                  |
| OFFCOMBTOPS         | Combined local/toll with TOPS                                   |

**Related functional groups**

The following functional groups are associated with OM group OFZ2NET2:

- Carrier VoIP North America
- Carrier VoIP International

## Registers

The following table lists the registers associated with OM group OFZ2NET2 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OFZ2NET2

| Register name           | Measures                                       |
|-------------------------|------------------------------------------------|
| <a href="#">DPTRC</a>   | Dynamic packet trunk (DPT) reservation control |
| <a href="#">MDPD</a>    | Machine-dialed partial dials                   |
| <a href="#">MDPSIG</a>  | Machine-dialed permanent signal                |
| <a href="#">NCKTBN</a>  | No circuit business network trunks             |
| <a href="#">NCKTID</a>  | No circuit inward dial trunks                  |
| <a href="#">NCKTIM</a>  | No circuit inter-machine trunks                |
| <a href="#">NCKTIT</a>  | No circuit inter-toll trunks                   |
| <a href="#">NCKTLT</a>  | No circuit local tandem trunks                 |
| <a href="#">NCKTOF</a>  | No circuit off-net trunks                      |
| <a href="#">NCKTON</a>  | No circuit connect trunks                      |
| <a href="#">NCKTOSC</a> | No service circuit trunks                      |
| <a href="#">NCKTOT</a>  | No circuit other trunk                         |
| <a href="#">NCKTRT</a>  | No circuit trunks                              |
| <a href="#">NCKTTC</a>  | No circuit toll completing trunks              |

### NCKTBN

**Register type**

Peg

**Description**

NCKTBN counts calls on the associated bearer network that route to GNCT because a Meridian Digital Centrex (MDC) is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTID****Register type**

Peg

**Description**

NCKTID counts calls on the associated bearer network that route to GNCT because a direct inward dial or direct outward dial trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTIM****Register type**

Peg

**Description**

NCKTIM counts calls on the associated bearer network that route to GNCT because a circuit inter-machine trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTIT****Register type**

Peg

**Description**

NCKTIT counts calls on the associated bearer network that the system routes to GNCT because an inter-toll trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTLT****Register type**

Peg

**Description**

NCKTLT counts calls on the associated bearer network that the system routes to GNCT because a local tandem trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTOF****Register type**

Peg

**Description**

NCKTOF counts calls on the associated bearer network that route to GNCT because an off-net or direct dial trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138



**NCKTON****Register type**

Peg

**Description**

NCKTON counts calls on the associated bearer network that route to GNCT because a dedicated access or mobile telephone exchange trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTOT****Register type**

Peg

**Description**

NCKTOT counts calls on the associated bearer network that route to GNCT because one of the following trunk types is not available:

- test line or desk
- maintenance trunks
- AV101

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTRT****Register type**

Peg

**Description**

NCKTRT counts calls on the associated bearer network that route to GNCT because one of the following trunk types is not available:

- 0+/0- tandem to TOPS
- outgoing to AMR2 or CAMA
- outgoing local
- recording completing outgoing
- TOPS outgoing

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTTC****Register type**

Peg

**Description**

NCKTTC counts calls on the associated bearer network that route to GNCT because a toll completing trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**NCKTOSC****Register type**

Peg

**Description**

NCKTOSC counts calls on the associated bearer network that route to GNCT because automatic number announcement or automatic intercept trunk is not available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

ATB100, LINE138, TRK138

**MDPD****Register type**

Peg

**Description**

MDPD counts calls on the associated bearer network that route to partial dial treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK138, TRK182

**MDPSIG****Register type**

Peg

**Description**

MDPSIG counts machine-dialed calls on the associated bearer network that route to permanent signal treatment.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

TRK115, TRK117, TRK138, TRK183

**DPTRC****Register type**

Peg

**Description**

DPTRC counts the number of dynamic packet trunk (DPT) terminal requests that are blocked because of DPT bandwidth reservation.

**Associated registers**

None

**Extension registers**

DPTRC2

**Associated logs**

None

## OGTQMS

### Description

Outgoing trunk queue management system (QMS)

OGTQMS records the number of times an operator enters outgoing trunk (OGT) keystroke actions.

The following table lists the key and info fields associated with OM group OGTQMS.

| Key field | Info field                                                                              |
|-----------|-----------------------------------------------------------------------------------------|
| none      | key type {OT, CT4Q, ASST, LANG, DUALLANG} and key label as datafilled in table TQOGTKEY |

### Related functional groups

The QMS functional group is associated with OM group OGTQMS.

### Registers

The following table lists the registers associated with OM group OGTQMS and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OGTQMS

| Register name           | Measures        |
|-------------------------|-----------------|
| <a href="#">KEYHITS</a> | OGTQMS key hits |

#### KEYHITS

##### Register type

Peg

##### Description

OGTQMS key hits

KEYHITS is incremented each time the operator enters OGT keystroke actions.

##### Associated registers

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## OHBTDTU

### Description

Off-Hook Balance Test Digital Test Unit (OHBTDTU)

The OM group OHBTDTU monitors the following:

- the number of digital test units (DTU) available at midnight
- the number of DTUs available at the time of any OMSHOW request
- the number of DTU seizures after midnight

The OM group OHBTDTU provides one tuple. The following table lists the key and info fields associated with OM group OHBTDTU.

| Key field | Info field |
|-----------|------------|
| none      | none       |

### Related functional groups

The Base Line Maintenance functional group associates with the OM group OHBTDTU.

### Registers

The following table lists the registers associated with OM group OHBTDTU and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OHBTDTU

| Register name          | Measures                                  |
|------------------------|-------------------------------------------|
| <a href="#">DTUMID</a> | Total DTUs available at midnight (DTUMID) |
| <a href="#">DTUNOW</a> | Total DTUs available now (DTUNOW)         |
| <a href="#">DTUSZD</a> | Total DTU seizures in the day (DTUSZD)    |

#### DTUMID

##### Register type

Peg

##### Description

Total DTUs available at midnight (DTUMID)

Register DTUMID has one field. This register increases at midnight to give the number of available DTUs at midnight in the OHBTADMN DTU list.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DTUNOW****Register type**

Peg

**Description**

Total DTUs available now (DTUNOW)

Register DTUNOW has one field. This register increases each time a DTU is deleted or a DTU is added to the available list in table OHBTADMN. When the user makes an OMSHOW request, the register gives the number of available DTUs.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DTUSZD****Register type**

Peg

**Description**

Total DTU seizures in the day (DTUSZD)

Register DTUSZD has one field. This register increases when the system seizes a DTU for an OHBT.



**Associated registers**

The associated registers for DTUSZD are as follows:

- Registers ORIG, TOA and TOS in OM OHBTTYPE increase when the system performs an OHBT and seizes a DTU. The number of seizures in register DTUTOTAL will equal the number of test performed fields of these three registers.
- Register DTUSZD in OM OHBTRES increases when the system seizes a DTU within the hour.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## OHBTRES

### Description

Off-hook balance testing (OHBT)

The OHBT digital test unit (DTU) resource utilization (OHBTTRES) OM group monitors the following:

- the number of completed OHBT tests
- the number of OHBT tests that failed to complete because of a lack of DTU availability

The OM group OHBTRES provides 24 tuples. The following table lists the key and info fields associated with OM group OHBTRES.

| Key field | Info field |
|-----------|------------|
| none      | none       |

### Related functional groups

BASE Line Maintenance

### Registers

The following table lists the registers associated with OM group OHBTRES and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OHBTRES

| Register name           | Measures                               |
|-------------------------|----------------------------------------|
| <a href="#">DTUSZD</a>  | Total DTU seizures in the day (DTUSZD) |
| <a href="#">SZDFAIL</a> | Seize failed (SZFAIL)                  |

#### DTUSZD

##### Register type

Peg

##### Description

Total DTU seizures in the day (DTUSZD)

The DTUSZD register increases if the OHBT test seized a DTU in the hour.

**Associated registers**

Registers ORIG, TOA and TOS in OM group OHBTTYPE increase when an OHBT test seizes a DTU. The seizures in register DTUTOTAL equal the number of tests that the system performs for these three registers.

Register [SZDFAIL](#) increases when a DTU is not available to be seized.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SZDFAIL****Register type**

Peg

**Description**

Seize failed (SZFAIL)

Register SZDFAIL increases every hour in the correct field when an OHBT cannot seize the DTU. The hour of the day determines the field that increases.

**Associated registers**

Registers ORIG, TOA and TOS in OMOHBTTYPE increase when an OHBT does not recommend a balance network configuration.

Register [DTUSZD](#) increases when an OHBT seizes a DTU.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates log number 602 when an OHBT cannot seize the DTU.

## OHBTTYPE

### Description

Off-Hook Balance Test Results per test type (OHBTTYPE)

The OM group OHBTTYPE monitors the following:

- the number of Off-Hook Balance Tests (OHBT) that the system performs
- the number of OHBTs that fail to complete

The OM group OHBTTYPE provides two tuples. The following table lists the key and info fields associated with OM group OHBTTYPE.

| Key field | Info field |
|-----------|------------|
| none      | none       |

### Related functional groups

The Base Line Maintenance functional group associates with the OM group OHBTTYPE.

### Registers

The following table lists the registers associated with OM group OHBTTYPE and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OHBTTYPE

| Register name        | Measures                              |
|----------------------|---------------------------------------|
| <a href="#">ORIG</a> | Originating OHBT (ORIG)               |
| <a href="#">TOA</a>  | Terminating test on answer OHBT (TOA) |
| <a href="#">TOS</a>  | Terminating test on silence (TOS)     |

#### ORIG

##### Register type

Peg

##### Description

Originating OHBT (ORIG)

Register ORIG register has two fields.

- The first field increases when the system performs an originating OHBT.
- The second field increases when an OHBT does not recommend a network balance configuration.

### **Associated registers**

The associated registers for ORIG are as follows:

- Register DTUSZD in OM group OHBTRES increases if an OHBT seized a DTU in the hour.
- Register DTUTOTAL in OM group OHBTDTU increases if an OHBT seized a DTU.
- Register SZDFAIL in OM group OHBTRES increases if an OHBT failed to seize a DTU because no DTU was available.

### **Extension registers**

There are no extension registers.

### **Associated logs**

The associated logs for ORIG are as follows:

- The system generates log 600 when an OHBT recommends the current network balance configuration.
- The system generates log 601 when an OHBT recommends a new network balance configuration.
- The system generates log 602 when an OHBT fails to complete and does not recommend a network balance configuration.

## **TOA**

### **Register type**

Peg

### **Description**

The associated registers for TOA are as follows:

- Register DTUTOTAL in OM group OHBTDTU increases if an OHBT seized a DTU.
- Register DTUSZD in OM group OHBTRES increases if an OHBT seized a DTU in the hour.
- Register SZDFAIL in OM group OHBTRES increases if an OHBT failed to seize a DTU because no DTU was available.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

The associated logs for TOA are as follows:

- The system generates log 600 when an OHBT recommends the current network balance configuration.
- The system generates log 601 when an OHBT recommends a new network balance configuration.
- The system generates log 602 when an OHBT fails to complete and does not recommend a network balance configuration.

**TOS****Register type**

Peg

**Description**

Terminating test on silence (TOS)

Register TOS has two fields. The first field increases when the system performs a terminating OHBT. The second field increases when an OHBT does not recommend a network balance configuration.

**Associated registers**

The associated registers for TOS are as follows:

- Register DTUTOTAL in OM group OHBTDTU increases if an OHBT seized a DTU.
- Register DTUSZD in OM group OHBTRES increases if an OHBT seized a DTU in the hour.
- Register SZDFAIL in OM group OHBTRES increases if an OHBT does not seize a DTU because no DTU was available.

**Extension registers**

There are no extension registers.

**Associated logs**

The associated logs for TOS are as follows:

- The system generates log 600 when an OHBT recommends the current network balance configuration
- The system generates log 601 when an OHBT recommends a new network balance configuration.
- The system generates log 602 when an OHBT fails to complete and does not recommend a network balance configuration.

## OHQCBQCG

### Description

Off-hook queuing and call back queuing per customer group (OHQCBQCG)

The OM group OHQCBQCG provides information about the following integrated business network (IBN) features for a customer group:

- Off-hook Queuing (OHQ)
- Call Back Queuing (CBQ)

If the system cannot complete a call from a station or an incoming trunk, the calling party can wait off-hook for an idle trunk. The system cannot complete the call because an idle outgoing trunk in the route set is not available. The system gives the caller off-hook queue tone. The system places the caller in a queue that associates with the outgoing trunk group. When an idle outgoing trunk becomes available, the system completes the call.

The CBQ feature activates when a caller encounters an all-trunks-busy (ATB) condition. The system places the call in a queue that associates with the trunk group. The system informs the caller when a trunk becomes available. The system uses the number dialed earlier to complete the call.

The OHQ and CBQ features are assigned in table NCOS.

Use of either OHQ or CBQ features can indicate that more trunks than necessary are on a specified route.

The OM group OHQCBQCG provides one tuple for each customer group. The following table lists the key and info fields associated with OM group OHQCBQCG.

| Key field | Info field                                                                 |
|-----------|----------------------------------------------------------------------------|
| none      | OMIBNGINGO identifies the CUSTNAME of the customer group in table CUSTENG. |

Parameter AVG\_#\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups that are involved in OHQ and CBQ.



Parameter NUMOHCQBQTRANSBLKS in table OFCENG specifies the number of transaction blocks in use for OHQ and CBQ.

Parameter NO\_OF\_FTR\_CONTROL\_BLKs in table OFCENG specifies the number of feature control blocks in use for OHQ and CBQ.

Parameter NO\_OF\_FTR\_DATA\_BLKs in table OFCENG specifies the number of FTRQ2 feature data blocks in use for OHQ and CBQ.

Parameter FTRQAGENTS in table OFCENG specifies the number of agents that can have the CBQ feature at any given time.

Parameter FTRQ2WAREAS in table OFCENG specifies the number of FTRQ2 word areas required for the engineering interval that associates with CBQ.

## Related functional groups

The OM group OHQCBQCG associates with the IBN Integrated Business Network functional group.

## Registers

The following table lists the registers associated with OM group OHQCBQCG and what they measure. For a description of a register, click on the register name.

### Registers for OM group OHQCBQCG

| Register name            | Measures                                   |
|--------------------------|--------------------------------------------|
| <a href="#">CBQDEACT</a> | Call back queuing deactivations            |
| <a href="#">CBQDELT</a>  | Call back queuing deletions                |
| <a href="#">CBQOK</a>    | Call back queuing okay                     |
| <a href="#">CBQOVFL</a>  | Call back queuing overflows                |
| <a href="#">CBQOVWRT</a> | Call back queuing overwrites               |
| <a href="#">CBQPPT</a>   | Call back queuing priority promotion timer |
| <a href="#">CBQRAT</a>   | Call back queuing route advance timer      |
| <a href="#">OHQABN</a>   | Off-hook queuing abandons                  |
| <a href="#">OHQBLOCK</a> | Off-hook queuing blockages                 |

**Registers for OM group OHQCBQCG**

| Register name            | Measures                   |
|--------------------------|----------------------------|
| <a href="#">OHQOFFER</a> | Off-hook queuing offers    |
| <a href="#">OHQOVFL</a>  | Off-hook queuing overflows |

**CBQDEACT****Register type**

Peg

**Description**

Call back queuing deactivations (CBQDEACT)

Register CBQDEACT increases when the system cancels a CBQ request. To cancel the CBQ request the subscriber dials the CBQ deactivation code while the CBQ is active. To cancel the CBQ request the subscriber can also press the CBQ key on a business set while CBQ is active.

**Associated registers**

Register OHQCBQRT\_RTCBQDEA increases when the system cancels a CBQ request. To cancel the CBQ request the subscriber dials the CBQ deactivation code while the CBQ is active. To cancel the CBQ request, the subscriber can also press the CBQ key on a business set while CBQ is active.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CBQDELT****Register type**

Peg

**Description**

Call back queuing deletions (CBQDELT)

Register CBQDELT increases when the system deletes a CBQ request.

The system deletes the request for one of the following reasons:

- the originator did not answer the recall
- the system removes the line from service
- the system cancels the CBQ option.

**Associated registers**

Registers OHQCBQRT\_RTCBQDELT counts deletions of CBQ for each route.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CBQOK****Register type**

Peg

**Description**

Call back queuing okay (CBQOK)

Register CBQOK increases when the system completes a CBQ request and the originator answers the recall ringback.

**Associated registers**

Register OHQCBQRT\_RTCBQOK counts successful completions of CBQ for each route.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CBQOVFL****Register type**

Peg

**Description**

Call back queuing overflows (CBQOVFL)

Register CBQOVFL increases when the system cannot complete a CBQ request because of not enough software resources.

Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the number of transaction blocks used in an office for both OHQ and CBQ. Parameter AVG\_#\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups involved in an OHQ/CBQ call.

If transaction blocks are not available during a CBQ request, the system denies the request.

**Associated registers**

Register OHQCBQRT\_RTCBQOVF counts CBQ requests for each route that the system cannot complete because there are not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to treatment after being call processing busy.

**CBQOVWRT****Register type**

Peg

**Description**

Call back queuing overwrites (CBQOVWRT)

Register CBQOVWRT increases when other CBQ or ring again (RAG) requests overwrite a CBQ request. This procedure occurs when the caller has a CBQ request pending and the caller activates CBQ on another call. The caller must activate CBQ on this call before completion of the original request.

To overwrite a CBQ request on a business set, cancel a remaining CBQ request. Cancellation of the CBQ request must occur before activation of the feature can occur on a different call.

**Associated registers**

Register OHQCBQRT\_RTCBQOVW counts CBQ requests for each route that other CBQ or RAG requests overwrite.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CBQPPT****Register type**

Peg

**Description**

Call back queuing priority promotion timer (CBQPPT)

Register CBQPPT increases when the CBQ priority promotion timer for a call ends and the CBQ priority promotion of the call occurs.

The queue priority promotion time is the maximum time a station queues at a specified level in the priority ordered queue. The CBQ starting priority can be one of four levels. The CBQ maximum priority is the highest level the station can achieve in the priority-ordered queue. The request qualifies for priority promotion when the starting priority is less than the maximum priority. When the priority promotion tone expires, the starting priority is less than the maximum priority.

The CBQ priority promotion timer appears in table CUSTSTN. The CBQ starting priority and maximum priority are in table NCOS.

**Associated registers**

Register OHQCBQRT\_RTCBQPPT increases when the CBQ priority promotion timer for a call ends and the CBQ priority promotion of the call occurs.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CBQRAT****Register type**

Peg

**Description**

Call back queuing route advance timer (CBQRAT)

Register CBQRAT increases when the CBQ route advance timer for a CBQ request ends. Qualify the CBQ request for CBQ route advance timing.

The CBQ route advance timer prevents long delays during heavy traffic periods. At the start, a request to queue a call back on a route with a

lower cost occurs. When the timer expires, the CBQ request qualifies for completion on routes with both higher and lower costs.

To apply this feature to stations in a customer group, enter the field CBQRAT in table CUSTSTN.

**Associated registers**

Register OHQCBQRT\_RTCBQRAT increases when the CBQ route advance timer for a CBQ request ends.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**OHQABN****Register type**

Peg

**Description**

Off-hook queuing abandons (OHQABN)

Register OHQABN increases when the calling party abandons the OHQ attempt before the system can complete the request. This register counts calls that one of the following methods abandons:

- go on-hook to terminate the OHQ attempt
- flash and going on-hook to activate CBQ
- activate the CBQ feature on a business set and going on-hook

**Associated registers**

Register OHQCBQRT\_RTOHQABN increases when the calling party abandons the OHQ attempt before the system completes the request.

**Extension registers**

There are no extension registers.

**Associated logs**

Log LINE106 increases when the system encounters problems during dial pulse reception.

Log LINE108 increases when the system encounters problems during Digitone reception.

Log TRK114 increases when the following events occur:

- the system encounters problems during dial pulse reception for an incoming call over a trunk
- the system does not determine the call destination Log

TRK116 increases when the following events occur:

- the system encounters problems during multi frequency reception for an incoming call over a trunk
- the system does not determine the call destination

Log TRK162 increases when the system encounters problems during outpulsing of a trunk-to-trunk or line-to-line call. The outpulsing occurs while digital multi frequency signaling is in use.

## **OHQBLOCK**

### **Register type**

Peg

### **Description**

Off-hook queuing blockages (OHQBLOCK)

Register OHQBLOCK increases when the system blocks an OHQ request because the system cannot complete the request before a specified wait timeout. Entries for the wait timeout appear in table IBNRTE.

Register OHQBLOCK increases when a likelihood test fails. The likelihood test decides if a call can be assigned an idle trunk within the wait timeout.

### **Associated registers**

Register TRMT1\_GNCT increases when the system routes a call that failed the likelihood test to the treatment.

For each route, register OHQCBQRT\_RTOHQBLOCK increases when the system blocks an OHQ request. The system blocks the OHQ request because the system cannot complete the request in a specified timeout period.

### **Extension registers**

There are no extension registers.

### **Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a specified numbering plan area (NPA). The system

generates ATB100 when the system blocks an attempt to seize a trunk to a specified central office (CO). The call advances to another route.

## **OHQOFFER**

### **Register type**

Peg

### **Description**

Off-hook queuing offers (OHQOFFER)

Register OHQOFFER increases when the system offers OHQ to a user because trunks are not available on the preferred route.

### **Associated registers**

Register OHQCBQRT\_RTOHQOFR increases when the system blocks an OHQ request because the system cannot complete the request in a specified timeout period. The register increases for each route.

### **Extension registers**

There are no extension registers.

### **Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a specified NPA or CO. The call advances to another route.

## **OHQOVFL**

### **Register type**

Peg

### **Description**

Off-hook queuing overflows (OHQOVFL)

Register OHQOVFL increases when the system cannot complete an OHQ request because of not enough software resources.

Parameter AVG\_#\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups that will be involved in an OHQ or CBQ call.

Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the number of transaction blocks that an office can use for OHQ and CBQ.



**Associated registers**

For each route, register OHQCBQRT\_RTOHQOVFL increases when the system cannot complete an OHQ request because of not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

Logs LINE138 and TRK138 increase when the system routes a call to a treatment because a log is call-processing busy.

---

## OHQCBQR2

---

### Description

Off-hook queuing and call back queuing for table IBNRT2 routes (OHQCBQR2)

The OM group OHQCBQR2 provides information for each group in table IBNRT2 on the following:

- Meridian Digital Centrex (MDC) features
- off-hook queuing (OHQ)
- all back queuing (CBQ)

If the system cannot complete a call from a station or an incoming trunk, the calling party can wait off-hook for an idle trunk. The system cannot complete the call because of an idle outgoing trunk in the route set is not available. The system gives the caller an off-hook queue tone. The system places the caller in a queue that associates with the outgoing trunk group. The system completes the call when an idle outgoing trunk becomes available.

The CBQ feature becomes active when a caller encounters an all-trunks-busy (ATB) condition. The system places the call in a queue that associates with the trunk group. The system informs the caller when a trunk becomes available. The system uses the number dialed earlier to complete the call.

The OHQ and CBQ features are assigned in table NCOS.

The OHQCBQR2 contains 11 registers that count:

- the CBQ requests that the system cancels
- the CBQ requests that the system delete
- the CBQ requests that the system completes
- the CBQ requests that the system cannot complete because of not enough software resources
- the CBQ requests that other CBQ or ring again requests overwrite
- the times the CBQ priority promotion timer for a call comes to an end and the times the CBQ priority promotion occurs
- the number of times the CBQ route advance timer for a CBQ request comes to an end
- the OHQ attempts that the calling party abandons

- the OHQ requests that the system blocks
- the number of times the system offers OHQ to a user because trunks are not available on the desired route
- the OHQ requests that the system cannot complete because of not enough software resources

The OM group OHQCBQR2 provides one tuple for each route in table IBNRT2. The following table lists the key and info fields associated with OM group OHQCBQR2.

| Key field | Info field                                                 |
|-----------|------------------------------------------------------------|
| none      | OM_IBN_RT2_INFO. The route number appears in table IBNRT2. |

## Related functional groups

There are no related functional groups.

## Registers

The following table lists the registers associated with OM group OHQCBQR2 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OHQCBQR2

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">R2CBQDEA</a> | Route call back queuing deactivations            |
| <a href="#">R2CBQDEL</a> | Route call back queuing deletions                |
| <a href="#">R2CBQOK</a>  | Route call back queuing okay                     |
| <a href="#">R2CBQOVE</a> | Route call back queuing overflows                |
| <a href="#">R2CBQOWR</a> | Route call back queuing overwrites               |
| <a href="#">R2CBQPPT</a> | Route call back queuing priority promotion timer |
| <a href="#">R2CBQRAT</a> | Route call back queuing route advance timer      |
| <a href="#">R2CHQABN</a> | Route off-hook queuing abandons                  |

**Registers for OM group OHQCBQR2**

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">R2CHQBLK</a> | Route off-hook queuing blockages |
| <a href="#">R2CHQOFR</a> | Route off-hook queuing offers    |
| <a href="#">R2CHQOVF</a> | Route off-hook queuing overflows |

**R2CBQDEA****Register type**

Peg

**Description**

Route call back queuing deactivations (R2CBQDEA)

Register R2CBQDEA counts CBQ requests that the system cancels. The system cancels these requests when the subscriber dials the CBQ deactivation code.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDEACT counts CBQ requests that the system cancels. The system cancels these requests when the subscriber dials the CBQ deactivation code while CBQ is active. The system also cancels these requests when the subscriber presses the CBQ key on a business set while CBQ is active.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CBQDEL****Register type**

Peg

**Description**

Route call back queuing deletions (R2CBQDEL)

Register R2CBQDEL counts CBQ requests that the system deletes.

The system deletes the request for one of the following reasons:

- the originator does not answer the recall
- the system removes the line from service

- the system deactivates the CBQ option
- the system removes the CBQ option from the line

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDELT counts CBQ requests that the system deletes.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CBQOK****Register type**

Peg

**Description**

Route call back queuing okay (R2CBQOK)

Register R2CBQOK increases when the system completes a CBQ request and the originator answers the recall ringback.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOK increases when the system completes a CBQ request and the originator answers the recall ringback.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CBQOVF****Register type**

Peg

**Description**

Route call back queuing overflows (R2CBQOVF)

Register R2CBQOVF counts CBQ requests that the system cannot complete. The system cannot complete the requests because of not enough software resources.

Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the number of transaction blocks that can be used in an office for OHQ and CBQ.

Parameter AVG\_#\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups involved in an OHQ/CBQ call.

If transactions are not available during a CBQ request, the system denies the request.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOVFL counts CBQ requests that the system cannot complete because of not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

**R2CBQOWR****Register type**

Peg

**Description**

Route call back queuing overwrites (R2CBQOWR)

Register R2CBQOWR counts CBQ requests that other CBQ or ring again (RAG) requests overwrite. This procedure occurs when the following occur:

- the caller has a CBQ request that is pending
- the caller activates CBQ on another call before the system completes the original request

Register R2CBQOWR increases when a single line set dials an access code.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOVWRT counts CBQ requests that other CBQ or RAG requests overwrite.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CBQPPT****Register type**

Peg

**Description**

Route call back queuing priority promotion timer (R2CBQPPT)

Register R2CBQPPT increases when the CBQ priority promotion timer for a call comes to an end. Register R2CBQPPT increases when CBQ priority promotion of the call occurs.

The queue priority promotion time is the maximum time a station will be queued at a specified level in the priority-ordered queue. The CBQ starting priority can be one of four levels. The CBQ maximum priority is the highest level the station can reach in the priority-ordered queue. The request qualifies for priority promotion when the starting priority is less than the maximum priority. When the priority promotion timer expires, the starting priority is less than the maximum priority.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQPPT increases when the CBQ0 priority promotion timer for a call comes to an end. This register increases when CBQ priority promotion of the call occurs.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CBQRAT****Register type**

Peg

**Description**

Route call back queuing route advance timer (R2CBQRAT)

Register R2CBQRAT increases when the CBQ route advance timer for a CBQ request comes to an end. Qualify the CBQ request for CBQ route advance timing.

The system uses the CBQ route advance timer to prevent delays during heavy traffic periods. At the start, the system makes a request to queue

a call back on a route with a lower cost. When the timer expires, the system can complete the CBQ request on routes with higher and lower costs.

Entries for the field CBQRAT must appear in table CUSTSTN for this feature to apply to stations.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQRATRT increases when the CBQ route advance timer for a CBQ request comes to an end.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R2CHQABN****Register type**

Peg

**Description**

Route off-hook queuing abandons (R2CHQABN)

Register R2CHQABN increases when the calling party abandons an OHQ attempt before the system completes the procedure.

Register R2OHQABN counts calls that one of the following methods abandons:

- go on-hook to terminate the OHQ attempt
- dial call back queue access code
- flash switch-hook, dialing CBQ access code, and go on-hook to activate CBQ
- activate the CBQ feature on a business set and go on-hook

**Associated registers**

For a customer group, register OHQCBQCG\_OHQABN increases when the calling party abandons an OHQ attempt before the system completes the procedure.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



**R2CHQBLK****Register type**

Peg

**Description**

Route off-hook queuing blockages (R2CHQBLK)

Register R2CHQBLK increases when the system blocks an OHQ request. The system blocks the request because the system cannot complete the OHQ request before a specified wait timeout occurs. Entries for the timeout period are in table INBRTE2.

Register R2CHQBLK also increases when a likelihood test fails. A likelihood test determines the assignment of a call to an idle trunk within the wait timeout period.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQBLOCK increases when the system blocks an OHQ request. The system blocks the request because the system cannot complete the OHQ request before a specified wait timeout period.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a exact numbering plan area (NPA). The system also blocks an attempt to seize a trunk to a exact central office (CO). The call advances to another route.

**R2CHQOFR****Register type**

Peg

**Description**

Route off-hook queuing offers (R2CHQOFR)

Register R2CHQOFR increases when the system offers OHQ to a user because trunks are not available on the desired route.

**Associated registers**

Register OHQCBQCG\_OHQOFFER increases when the system offers OHQ to a user because trunks are not available on the desired route. The register increases for a customer group.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a specified numbering plan area (NPA). The system generates this log when the system blocks an attempt to seize a trunk to a specified central office (CO). The call advances to another route.

**R2CHQOVF****Register type**

Peg

**Description**

Route off-hook queuing overflows (R2CHQOVF)

Register R2CHQOVF counts OHQ requests that the system cannot complete because of not enough software resources.

Parameter AVG\_#\_TGS\_PER\_OHBCQCALL in table OFCENG specifies the average number of trunk groups that are involved in an OHQ or CBQ call. Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the number of transaction blocks that can be used in an office for OHQ and CBQ.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOVFL counts OHQ requests that the system cannot complete because of not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

---

## OHQCBQR3

---

### Description

Off-hook queuing and call back queuing for table IBNRT3 routes (OHQCBQR3)

The OM group OHQCBQR3 provides information for each route in table IBNRT3 on the following:

- Meridian Digital Centrex (MDC) features
- off-hook queuing (OHQ)
- call back queuing (CBQ)

If the system cannot complete a call from a station or an incoming trunk, the calling party can wait off-hook for an idle trunk. The system cannot complete a call because an idle outgoing trunk in the route set is not available. The system gives the caller an off-hook queue tone. The system places the caller in a queue that associates with the outgoing trunk group. When an idle outgoing trunk becomes available, the system completes the call.

The CBQ feature activates when a caller encounters an all-trunks-busy (ATB) condition. The system places the call in a queue that associates with the trunk group. When a trunk becomes available, the system informs the caller when a trunk becomes available. The system uses the number dialed earlier to complete the call.

OHQ and CBQ features are assigned in table NCOS. OHQCBQR3 contains 11 registers that count:

- the CBQ requests that the system cancels
- the CBQ requests that the system deletes
- the CBQ requests that the system completes
- the CBQ requests that the system cannot complete because there are not enough software resource
- the CBQ requests that are other CBQ or ring again requests overwrite
- the times the CBQ priority promotion timer for a call elapses and the CBQ priority promotion of the call occurs
- the times the CBQ route advance timer for a CBQ request elapses
- the OHQ attempts that the calling party abandons
- the OHQ requests that the system blocks

- the times the system offers OHQ to a user because trunks are not available on the desired route
- the OHQ requests that cannot be completed because of not enough software resources

The OM group OHQCBQR3 provides one tuple for each route in table IBNRT3. The following table lists the key and info fields associated with OM group OHQCBQR3.

| Key field | Info field                                                |
|-----------|-----------------------------------------------------------|
| none      | OM_IBN_RT3_INFO the route number appears in table IBNRT3. |

## Related functional groups

There are no related functional groups.

## Registers

The following table lists the registers associated with OM group OHQCBQR3 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OHQCBQR3

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">R3CBQDEA</a> | Route call back queuing deactivations            |
| <a href="#">R3CBQDEL</a> | Route call back queuing deletions                |
| <a href="#">R3CBQOK</a>  | Route call back queuing okay                     |
| <a href="#">R3CBQOVF</a> | Route call back queuing overflows                |
| <a href="#">R3CBQOWR</a> | Route call back queuing overwrites               |
| <a href="#">R3CBQPPT</a> | Route call back queuing priority promotion timer |
| <a href="#">R3CBQRAT</a> | Route call back queuing route advance timer      |
| <a href="#">R3CHQABN</a> | Route off-hook queuing abandons                  |
| <a href="#">R3CHQBLK</a> | Route off-hook queuing blockages                 |

**Registers for OM group OHQCBQR3**

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">R3CHQOFR</a> | Route off-hook queuing offers    |
| <a href="#">R3CHQOVF</a> | Route off-hook queuing overflows |

**R3CBQDEA****Register type**

Peg

**Description**

Route call back queuing deactivations (R3CBQDEA)

Register R3CBQDEA counts the subscriber requests. These cancellations occur when the subscriber dials the CBQ deactivation code.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDEACT counts CBQ requests that the system cancels. These cancellations occur when the subscriber dials the CBQ deactivation code while CBQ is active. These cancellations can also occur when the subscriber presses the CBQ key on a business set while CBQ is active.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CBQDEL****Register type**

Peg

**Description**

Route call back queuing deletions (R3CBQDEL)

Register R3CBQDEL counts CBQ requests that the system deletes.

The system deletes the request for one of the following reasons:

- the originator does not answer the recall • the system removes the line
- the system deactivates the CBQ option
- the system removes CBQ option from the line

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDELT counts CBQ requests that the system deletes.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CBQOK****Register type**

Peg

**Description**

Route call back queuing okay (R3CBQOK)

Register R3CBQOK counts the number of times a CBQ request that the system completes. The register counts the times the originator answers the recall ringback.

**Associated registers**

Register OHQCBQCG\_CBQOK counts the times a customer group completes a CBQ request. This register also counts the number of times the originator answers the recall ringback.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CBQOVF****Register type**

Peg

**Description**

Route call back queuing overflows (R3CBQOVF)

Register R3CBQOVF counts CBQ requests that the system cannot complete because there are not enough software resources.

Parameter NUMOHCQBQTRANSBLKS in table OFCENG specifies the number of transaction blocks that can be used in an office for both OHQ and CBQ.

Parameter AVG\_#\_TGS\_PER\_OHCQBQCALL in table OFCENG specifies the average number of trunk groups involved in an OHQ/CBQ call.

If transaction blocks are not available during a CBQ request, the system denies the request.

#### **Associated registers**

For a customer group, register OHQCBQCG\_CBQOVFL counts CBQ requests that cannot complete because there are not enough software resources.

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

The system generates LINE138 and TRK138 when the system routes call to a treatment after being call processing busy.

### **R3CBQOWR**

#### **Register type**

Peg

#### **Description**

Route call back queuing overwrites (R3CBQOWR)

Register R3CBQOWR counts CBQ requests that other CBQ or RAG requests overwrite. This procedure occurs when the caller has a CBQ request pending and activates CBQ on another call. The caller must activate before the system completes original request.

Register R3CBQOWR increases when the subscriber dials an access code on a single line set.

#### **Associated registers**

For a customer group, register OHQCBQCG\_CBQOVWRT counts CBQ requests that other CBQ or RAG requests overwrite.

#### **Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CBQPPT****Register type**

Peg

**Description**

Route call back queuing priority promotion timer (R3CBQPPT)

Register R3CBQPPT counts the times the CBQ priority promotion timer for a call ends. This register also counts the times the CBQ priority promotion of the call occurs.

The queue priority promotion time is the maximum time a station will be queued at a specified level in the priority-ordered queue. The CBQ starting priority can be one of four levels. The CBQ maximum priority is the highest level that the station can reach in the priority-ordered queue. The request qualifies for priority promotion when the starting priority is less than the maximum priority. When the promotion timer expires, the starting priority is less than the maximum penalty.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQPPT increases when the CBQ priority promotion timer for a call ends. This register also increases when the CBQ promotion of the call occurs.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CBQRAT****Register type**

Peg

**Description**

Route call back queuing route advance timer (R3CBQRAT)

Register R3CBQRAT increases when the CBQ route advance timer for a CBQ request elapses. Qualify the CBQ request for CBQ route advance timing.

The system uses the CBQ route advance timer to prevent delays during heavy traffic periods. The system makes a request to queue a call back



on an inexpensive route. The system can make the CBQ request on both inexpensive and expensive routes when the timer expires.

Entries for the field CBQRAT must appear in table CUSTSTN for this feature to apply to stations.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQRATRT increases when the CBQ route advance timer for a CBQ request ends.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CHQABN****Register type**

Peg

**Description**

Route off-hook queuing abandons (R3CHQABN)

Register R3CHQABN counts the times that the calling party abandons an OHQ attempt. This occurs before the system completes the request.

Register R3CHQABN counts calls that one of the following methods abandons:

go on-hook to terminate the OHQ attempt

flash the switch hook, dials the call back queue access code, and go on-hook to activate CBQ

activate the CBQ feature on a business set and going on-hook

**Associated registers**

For a customer group, register OHQCBQCG\_OHQABN counts the number of times that the calling party abandons an OHQ attempt before the attempt is complete.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R3CHQBLK****Register type**

Peg

**Description**

Route off-hook queuing blockages (R3CHQBLK)

Register R3CHQBLK counts the times the system blocks an OHQ request. Blockage occurs when the system did not complete the OHQ request before a specified wait timeout period. The entries for the wait timeout period appear in table INBRTE2.

Register R3CHQBLK increases when a likelihood test fails. The likelihood test determines if the system can assign a call to an idle trunk within the wait timeout period.

**Associated registers**

Register OHQCBQCG\_OHQBLOCK counts the times a customer group blocks an OHQ request. The blockage happens because the system cannot complete the OHQ request before a specified wait timeout period.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a specified numbering plan area (NPA). The system also blocks an attempt to seize a trunk to a specified central office (CO). The system advances the call to another route.

**R3CHQOFR****Register type**

Peg

**Description**

Route off-hook queuing offers (R3CHQOFR)

Register R3CHQOFR counts the times that the system offers OHQ to a user. The system advances occurs because trunks are not available on the desired route.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOFFER counts the times the system offers OHQ to a user. The system offers OHQ to a user because trunks are not available on the desired route.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates log ATB100 when the system blocks an attempt to seize a trunk to a given NPA. The system also blocks an attempt to seize a trunk to a given central office (CO). The call advances to another route.

**R3CHQOVF****Register type**

Peg

**Description**

Route off-hook queuing overflows (R3CHQOVF)

Register R3CHQOVF counts OHQ requests that the system cannot complete because there are not enough software resources.

Parameter AVG\_#\_TGS\_PER\_OHBCQCALL in table OFCENG specifies the average number of trunk groups that the system will involve in an OHQ or CBQ call. Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the transaction blocks that the system can use in an office for both OHQ and CBQ.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOVFL counts OHQ requests that the system cannot complete because there are not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates logs LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

---

## OHQCBQR4

---

### Description

Off-hook queuing and call back queuing for table IBNRT4 routes (OHQCBQR4)

For each route in table IBNRT4, the OM group OHQCBQR4 provides information on the following:

- Meridian Digital Centrex (MDC) features
- off-hook queuing
- call back queuing (CBQ)

If a call from either a station or an incoming trunk cannot be completed the calling party may wait off-hook for an idle trunk. These calls are not completed because an idle outgoing trunk in the route set is not available. The caller first receives an off-hook queue tone. The system places the tone in a queue that the outgoing trunk group associates with. The call completes when an idle outgoing trunk becomes available.

The CBQ feature activates when a caller encounters an all trunks busy (ATB) condition. A queue associated with the trunk group places the call. The system informs the caller when a trunk becomes available and the call is completed using the number dialed earlier.

The OHQ and CBQ features are assigned in table NCOS.

OHQCBQR4 contains 11 registers that count:

- the CBQ requests that the system cancels
- the CBQ requests that the system deletes
- the CBQ requests that the system completes
- the CBQ requests that the system cannot complete because there are not enough software resources
- the CBQ requests all back queuing requests that other CBQ or ring again requests overwrite
- the times the CBQ priority promotion timer for a call ends and the priority promotion of the call occurs
- the times the CBQ route advance timer for a CBQ request ends
- off-hook queuing attempts that are abandoned by the calling part
- the OHQ requests that the system blocks

- the times the system offers OHQ to a user because trunks are not available on the desired route
- the OHQ requests that the system cannot complete because there are not enough software resources

The OM group OHQCBQR4 provides one tuple for each route in table IBNRT4. The following table lists the key and info fields associated with OM group OHQCBQR4.

| Key field | Info field                                              |
|-----------|---------------------------------------------------------|
| none      | OM_IBM RT4 INFO. Table IBNRT4 assigns the route number. |

## Related functional groups

There are no related functional groups.

## Registers

The following table lists the registers associated with OM group OHQCBQR4 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OHQCBQR4

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">R4CBQDEA</a> | Route call back queuing deactivations            |
| <a href="#">R4CBQDEL</a> | Route call back queuing deletions                |
| <a href="#">R4CBQOK</a>  | Route call back queuing okay                     |
| <a href="#">R4CBQOVE</a> | Route call back queuing overflows                |
| <a href="#">R4CBQOWR</a> | Route call back queuing overwrites               |
| <a href="#">R4CBQPPT</a> | Route call back queuing priority promotion timer |
| <a href="#">R4CBQRAT</a> | Route call back queuing route advance time       |
| <a href="#">R4CHQABN</a> | Route off-hook queuing abandons                  |
| <a href="#">R4CHQBLK</a> | Route off-hook queuing blockages                 |

**Registers for OM group OHQCBQR4**

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">R4CHQOFR</a> | Route off-hook queuing offers    |
| <a href="#">R4CHQOVF</a> | Route off-hook queuing overflows |

**R4CBQDEA****Register type**

Peg

**Description**

Route call back queuing deactivations (R4CBQDEA) Register R4CBQDEA counts CBQ requests that the system cancels. Cancellation occurs when the subscriber dials the CBQ deactivation code.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDEACT counts CBQ requests that the system cancels when the subscriber dials the CBQ deactivation code. When the subscriber presses the CBQ key on a business set while CBQ is active, the system can cancel CBQ requests.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CBQDEL****Register type**

Peg

**Description**

Route call back queuing deletions (R4CBQDEL)

Registers R4CBQDEL counts CBQ requests that the system deletes.

The system can delete the request for one of the following reasons:

- the originator does not answer the recall
- the system removes the line
- the system deactivates the CBQ option
- the system removes the CBQ option from the line

**Associated registers**

For a customer group, register OHQCBQCG\_CBQDELT counts CBQ requests that the system deletes.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CBQOK****Register type**

Peg

**Description**

Route call back queuing okay (R4CBQOK)

Register R4CBQOK counts the times that the system completes a CBQ request and the originator answers the recall ringback.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOK counts the number of times that the system completes a CBQ request and the originator answers the recall ringback.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CBQOVF****Register type**

Peg

**Description**

Route call back queuing overflows (R4CBQOVF)

Register R4CBQOVF counts CBQ requests that the system cannot complete because there are not enough software resources.

Parameters NUMOHCBQTRANSBLKS in table OFCENG specifies transaction blocks that can be used in an office because of both OHQ and CBQ.

The average number of trunk groups involved in an OHQ/CBQ call is specified by Parameter AVG\_#\_TGS\_PER\_OHCBQCALL in table OFCENG.

The system denies the request if transaction blocks are not available during a CBQ request.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOVFL counts CBQ requests that the system cannot complete because there are not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

**R4CBQOWR****Register type**

Peg

**Description**

Route call back queuing overwrites (R4CBQOWR)

Register R4CBQOWR counts the requests that other CBQ or ring again RAG requests overwrite. This occurs when the caller has a CBQ request that is pending. The caller activates CBQ on another call before the system completes the original request.

Register R4CBQOWR increases when a single line set dials an access code.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQOVWRT counts the CBQ or RAG requests that other CBQ or RAG requests overwrite.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



**R4CBQPPT****Register type**

Peg

**Description**

Route call back queuing priority promotion timer (R4CBQPPT)

Register R4CBQPPT increases when the CBQ priority promotion timer for a call ends and the CBQ priority promotion of the call occurs.

The queue priority promotion time is the maximum time a station will be queued at a given level in the priority-ordered queue. The CBQ starting priority can be one of four levels. The CBQ maximum priority is the highest level that the station can reach in the priority-ordered queue. The request qualifies for priority promotion when the starting priority is less than the maximum priority. When the priority promotion time expires, the starting priority is less than the maximum priority.

**Associated registers**

For customer group, register OHQCBQCG\_CBQPPT increases when the CBQ priority promotion timer for a call ends. This register also increases when the CBQ priority promotion of the call occurs.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CBQRAT****Register type**

Peg

**Description**

Route call back queuing route advance time (R4CBQRAT)

Route call back queuing route advance timer (R4CBQRAT) increases when the call back queuing (CBQ) route advance timer for a CBQ request ends. Qualify the CBQ request for CBQ route advance timing.

The system uses the CBQ route advance timer to prevent delays during heavy traffic periods. At the start, the system makes a request to queue a call back on an inexpensive route. The CBQ request can be completed on inexpensive routes when the timer expires.

Entries for the field CBQRAT appear in table CUSTSTN for stations to apply this feature.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQRATRT increases when the CBQ route advance timer for a CBQ request ends.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CHQABN****Register type**

Peg

**Description**

Route off-hook queuing abandons (R4CHQABN)

Register R4CHQABN counts the times the calling party abandons an OHQ attempt before the system completes the attempt.

Register R4CHQABN counts calls that one of the following methods abandons:

- go on-hook to terminate the OHQ attempt
- flash the switch hook, dial the call back queue access code, and going on-hook to activate CBQ
- activate the CBQ feature on a business set and going on-hook

**Associated registers**

For a customer group, register OHQCBQCG\_OHQABN counts the times the calling party abandons an OHQ attempt before the system completes the attempt.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**R4CHQBLK****Register type**

Peg

**Description**

Route off-hook queuing blockages (R4CHQBLK)

Register R4CHQBLK counts the times that the system blocks an OHQ request. This occurs because the system cannot complete the OHQ before a specified wait timeout period. Entries for the wait timeout are in table INBRTE2.

Register R4CHQBLK also increases when a likelihood test fails. The likelihood test determines if the system can assign an idle trunk to a call in the wait timeout period.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQBLOCK counts the times that an OHQ blocks a request. The blockage occurs because the system cannot complete the request before a specified wait timeout period.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to a given numbering plan area (NPA). The system also blocks an attempt to seize a trunk to a given central office (CO). The system advances the call to another route.

**R4CHQOFR****Register type**

Peg

**Description**

Route off-hook queuing offers (R4CHQOFR)

Register R4CHQOFR counts the times the system offers OHQ to a user. the system offers OHQ because trunks are not available on the desired route.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOFFER counts the number of times the system offers OHQ. The system offers OHQ because trunks are not available on the desired route.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk for one of the following:

- a given numbering plan area (NPA)
- or central office (CO)

The call advances to another route.

**R4CHQOVF****Register type**

Peg

**Description**

Route off-hook queuing overflows (R4CHQOVF)

Route off-hook queuing overflows (R4CHQOVF) counts OHQ requests that the system cannot because there are not enough software resources.

Parameter AVG\_#\_TGS\_PER\_OHBCQCALL in table OFCENG specifies the average number of trunk groups that will be involved in an OHQ or CBQ call. Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the transaction blocks that can be used in an office for both OHQ and CBQ.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOVFL counts OHQ requests that the system cannot complete because there are not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

## OHQCBQRT

### Description

Off-hook queuing and call back queuing per route (OHQCBQRT)

For each route, the OM group OHQCBQRT provides information on the integrated business network (IBN) features off-hook queuing (OHQ) and call back queuing (CBQ).

If the system cannot complete a call from a station or an incoming trunk, the calling party can wait off-hook for an idle trunk. The system cannot complete the call because an idle outgoing trunk in the route set is not available. The system caller gives an off-hook queue tone. The system places the call in a queue that associates with the outgoing trunk group. When an idle outgoing trunk becomes available, the system completes the call.

If a caller encounters an all trunks busy (ATB) condition, the call back queuing (CBQ) feature can be activated. The call is placed in a queue associated with the trunk group. When a trunk becomes available, the caller is informed and the call is completed using the number dialed earlier.

The OHQ and CBQ features are assigned in table NCOS.

If the registers show little use of either OHQ or CBQ features, there may be more trunks provided than necessary on a route.

The OM group OHQCBQRT provides information about the integrated business network (IBN) features off-hook queuing (OHQ) for a customer group. This OM group also provides information about the call back queuing (CBQ) for a customer group.

The following table lists the key and info fields associated with OM group OHQCBQRT.

| Key field | Info field                                              |
|-----------|---------------------------------------------------------|
| none      | OM_IBN_RTE_INFO. Table IBNRTE assigns the route number. |

Parameter AVG\_NUM\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups that involve OHQ and CBQ.

Parameter NUMOHCQBQTRANSBLKS in table OFCENG specifies the number of transaction blocks in use for OHQ and CBQ.

Parameter NO\_OF\_FTR\_CONTROL\_BLKES in table OFCENG specifies the number of feature control blocks in use for OHQ and CBQ.

Parameter NO\_OF\_FTR\_DATA\_BLKES in table OFCENG specifies the number of feature data blocks in use for OHQ and CBQ.

Parameter FTRQAGENTS in table OFCENG specifies the number of agents that can have the CBQ feature at a time.

Parameter FTRQ2WAREAS in table OFCENG specifies the number of FTRQ2 word areas requires the engineering interval associated with CBQ.

## Related functional groups

The IBN Integrated Business Network operating group associates with OM group OHQCBQRT.

## Registers

The following table lists the registers associated with OM group OHQCBQRT and what they measure. For a description of a register, click on the register name.

### Registers for OM group OHQCBQRT

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">RTCBQDEA</a> | Route call back queuing deactivations            |
| <a href="#">RTCBQDEL</a> | Route call back queuing deletions                |
| <a href="#">RTCBQOK</a>  | Route call back queuing okay                     |
| <a href="#">RTCBQOVF</a> | Route call back queuing overflows                |
| <a href="#">RTCBQOWR</a> | Route call back queuing overwrites               |
| <a href="#">RTCBQPPT</a> | Route call back queuing priority promotion timer |
| <a href="#">RTCBQRAT</a> | Route call back queuing route advance timer      |
| <a href="#">RTOHQABN</a> | Route off-hook queuing abandons                  |
| <a href="#">RTOHQBLK</a> | Route off-hook queuing blockages                 |

**Registers for OM group OHQCBQRT**

| Register name            | Measures                         |
|--------------------------|----------------------------------|
| <a href="#">RTOHQOFR</a> | Route off-hook queuing offers    |
| <a href="#">RTOHQOVF</a> | Route off-hook queuing overflows |

**RTCBQDEA****Register type**

Peg

**Description**

Route call back queuing deactivations (RTCBQDEA)

Register RTCBQDEA increases when the subscriber cancels a call back queuing (CBQ) request. To cancel a CBQ request the caller can dial the CBQ deactivation code. Press the CBQ key on a business set while CBQ is active.

**Associated registers**

The system increases OHQCBQCG\_CBQDEACT for a customer group when the user cancels a call back queuing (CBQ) request. To cancel the request, dial the CBQ deactivation code or press the CBQ key on a business set while CBQ is active.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTCBQDEL****Register type**

Peg

**Description**

Route call back queuing deletions (RTCBQDEL)

Register RTCBQDEL increases when the system deletes a call back queuing (CBQ) request.

The system can delete the request for one of the following reasons:

- the originator did not answer the recall
- the system line removed from service
- the system canceled CBQ option

**Associated registers**

For a customer group, OHQCBQCG\_CBQDELT increases when the system deletes a call back queuing (CBQ) request.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTCBQOK****Register type**

Peg

**Description**

Route call back queuing okay (RTCBQOK)

Register RTCBQOK increases when a call back queuing (CBQ) request completes correctly and the originator answers the recall ringback.

**Associated registers**

For a customer group, OHQCBQCG\_CBQOK increases when a call back queuing (CBQ) request completes correctly and the originator answers recall ringback.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTCBQOVF****Register type**

Peg

**Description**

Route call back queuing overflows (RTCBQOVF)



Register RTCBQOVF increases when a call back queuing (CBQ) request cannot complete because there are not enough software resources.

Parameter NUMOHCQBQTRANSBLKS in table OFCENG specifies the number of transaction blocks that an office can use for OHQ and CBQ.

Parameter AVG\_NUM\_TGS\_PER\_OHCQBQCALL in table OFCENG specifies the average number of trunk groups an OHQ/CBQ call involves.

The system denies the request if no transaction blocks are available during a CBQ request.

#### **Associated registers**

For a customer group, registers OHQCBQCG\_CBQOVFL increases when a call back queuing (CBQ) request cannot complete. The request cannot complete because there are not enough software resources.

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after the call was processing busy.

### **RTCBQOWR**

#### **Register type**

Peg

#### **Description**

Route call back queuing overwrites (RTCBQOWR)

Register RTCBQOWR increases when a call back queuing (CBQ) request or ring again (RAG) request overwrites a CBQ request. This overwrite occurs when the caller has a CBQ request pending and activates CBQ on another call. The caller activates CBQ before the original request completes.

To overwrite a CBQ request on a business set, cancel the CBQ request that remains before you activate the feature on a different call.

#### **Associated registers**

Register OHQCBQCG\_CBQOVWRT increases for a customer group when a CBQ request or a ring again (RAG) request overwrites a call back queuing (CBQ) request.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTCBQPPT****Register type**

Peg

**Description**

Route call back queuing priority promotion timer (RTCBQPPT)

Register RTCBQPPT increases when the call back queuing (CBQ) priority promotion timer for a call ends. Call back queuing priority promotion of the call occurs.

The queue priority promotion time is the maximum time a station can remain queued at a level in the priority ordered queue. The CBQ starting priority can be one of four levels. The CBQ maximum priority is the highest level in the priority ordered queue. If the starting priority is less than the maximum priority, the request qualifies for priority promotion when the timer ends.

**Associated registers**

For customer group, register OHQCBQCG\_CBQPPT increases when the call back queuing (CBQ) priority promotion timer for a call finishes. Call back queuing priority promotion of the call must occur for the register to increase.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTCBQRAT****Register type**

Peg

**Description**

Route call back queuing route advance timer (RTCBQRAT)

Register RTCBQRAT increases when the call back queuing (CBQ) route advance timer for a CBQ request finishes. The CBQ request must qualify for CBQ route advance timing.

The CBQ route advance timer prevents delays in heavy traffic periods. The system makes a request to queue a call back on a low cost route. Qualify the CBQ request to complete on inexpensive and expensive routes when the timer expires.

Enter the field CBQRAT in table CUSTSTN to apply this feature to stations.

**Associated registers**

For a customer group, register OHQCBQCG\_CBQRATRTCBQRAT increases when the call back queuing (CBQ) route advance timer for a CBQ request finishes.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RTOHQABN****Register type**

Peg

**Description**

Route off-hook queuing abandons (RTOPHQABN)

Register RTOHQABN increases when the calling party abandons an off-hook queuing (OHQ) attempt before the attempt completes. This register counts calls that the system abandons by one of the following methods:

- user goes on-hook to terminate the OHQ attempts
- user flashes and goes on-hook to activate CBQ
- user activates the CBQ feature on a business set and goes on-hook

**Associated registers**

For a customer group, register OHQCBQCG\_OHQABN increases when the calling party abandons an off-hook queuing (OHQ) attempt before completion.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE106 when dial pulse reception on a line has problems.

The system generates LINE108 when Digitone reception on a line has problems.

The system generates TRK114 when dial pulse reception for an incoming call over a trunk has problems. The system did not determine the call destination.

The system generates TRK116 when a multi-frequency reception for an incoming call over a trunk has problems. The system can not determine the call destination.

The system generates TRK162 when transmission of either a trunk-to-trunk has problems. The system also generates this log when a line-to-line call uses digital multi-frequency signaling.

## **RTOHQBLK**

### **Register type**

Peg

### **Description**

Route off-hook queuing blockages (RTOHQBLK)

Register RTOHQBLK increases when the system blocks an off-hook queuing (OHQ) request because it cannot complete before a specified wait timeout. The wait timeout appears in table IBNRTE.

Register RTOHQBLK increases when a likelihood test fails. The likelihood test determines if the system can assign a call to an idle trunk within the wait timeout.

### **Associated registers**

For a customer group, register OHQCBQCG\_OHQBLOCK increases for a customer when the system blocks an off-hook queuing (OHQ). The system blocks the request because the request cannot complete before a specified wait timeout.

### **Extension registers**

There are no extension registers.

### **Associated logs**

The system generates the ATB100 when the system blocks an attempt to seize a trunk to an exact numbering plan area (NPA) or central office (CO). The call advances to another route.

**RTOHQOFR****Register type**

Peg

**Description**

Route off-hook queuing offers (RTOHQOFR)

Register RTOHQOFR increases when the system offers off-hook queuing (OHQ) to a user because trunks are not available on the desired route.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOFFER increases when the system offers off-hook queuing (OHQ) to a user. The system offers OHQ to the user because no available trunks are present on the desired route.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates ATB100 when the system blocks an attempt to seize a trunk to an exact numbering plan area (NPA) or central office (CO). The call advances to another route.

**RTOHQOVF****Register type**

Peg

**Description**

Route off-hook queuing overflows (RTOHQOVF)

Register RTOHQOVF increases when an off-hook queuing (OHQ) request cannot complete because there are not enough software resources.

Parameter AVG\_NUM\_TGS\_PER\_OHCBQCALL in table OFCENG specifies the average number of trunk groups the system involves in an OHQ or CBQ call. Parameter NUMOHCBQTRANSBLKS in table OFCENG specifies the number of transaction blocks an office can use for both OHQ and CBQ.

**Associated registers**

For a customer group, register OHQCBQCG\_OHQOVFL RTOHQOVF increase for a customer group when an off-hook queuing (OHQ)

request cannot complete. The request cannot complete when there are not enough software resources.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates LINE138 and TRK138 when the system routes a call to a treatment after being call processing busy.

## ONI

### Description

Operator number identification (ONI)

The ONI provides information about centralized automatic message accounting (CAMA) calls that use operator number identification (ONI).

The ONI allows a CAMA operator on the line to receive the calling number. The CAMA operator enters the calling number in the CAMA equipment for billing purposes.

The OM group ONI provides one tuple for each office. The following table lists the key and info fields associated with OM group ONI.

| Key field | Info field                                                              |
|-----------|-------------------------------------------------------------------------|
| none      | CPOS_OM_INFO. Number of CAMA positions software-defined for the office. |

### Related functional groups

The Traffic Operator Position (TOPS) operating group associates with the OM group ONI.

### Registers

The following table lists the registers associated with OM group ONI and what they measure. For a description of a register, click on the register name.

#### Registers for OM group ONI

| Register name            | Measures                                         |
|--------------------------|--------------------------------------------------|
| <a href="#">ONIATT</a>   | Operator number identification attempts          |
| <a href="#">ONICHDLU</a> | Operator number identification calls handled use |
| <a href="#">ONIDELGT</a> | Operator number identification delegation        |
| <a href="#">ONIFDISC</a> | Operator number identification forced disconnect |

**Registers for OM group ONI**

| Register name            | Measures                                        |
|--------------------------|-------------------------------------------------|
| <a href="#">ONIMBU</a>   | Operator number identification manual busy use  |
| <a href="#">ONIMTCHC</a> | Operator number identification match check      |
| <a href="#">ONIOCCU</a>  | Operator number identification occupied         |
| <a href="#">ONIOVFL</a>  | Operator number identification overflow         |
| <a href="#">ONIQABAN</a> | Operator number identification queue abandon    |
| <a href="#">ONIQOCC</a>  | Operator number identification queue occupied   |
| <a href="#">ONIQOVFL</a> | Operator number identification queue overflow   |
| <a href="#">ONIQTOUT</a> | Operator number identification queue timed out  |
| <a href="#">ONISBU</a>   | Operator number identification system busy      |
| <a href="#">ONISZRS</a>  | Operator number identification seizures         |
| <a href="#">ONIWRGCA</a> | Operator number identification wrong code added |

**ONIATT****Register type**

Peg

**Description**

Operator number identification attempts (ONIATT)

Register ONIATT counts calls the system routes to CAMA positions that use ONI. Register ONIATT counts calls the system routes to CAMA positions that use remote operator number identification (RONI).

**Associated registers**

Register [ONIQABAN](#) counts calls abandoned in the CAMA call waiting queue.

Register ONIQOVFL counts CAMA calls the system routes to no service circuit (NOSC) treatment.

Register ONIQTOUT counts calls that wait in the CAMA call waiting queue. The system routes calls to a treatment after enough time.



Register ONISZRS counts calls that connect to a CAMA position. The operator at the position acknowledges the calls.

Register ONIATT contains the following:

- [ONISZRS](#)
- [ONIQOVFL](#)
- [ONIQABAN](#)
- [ONIQTOUT](#)
- calls assigned to a position

Calls are abandoned while waiting for operator acknowledgment.

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

There are no associated logs.

### **ONICHDLU**

#### **Register type**

Peg

#### **Description**

Operator number identification calls handled use (ONICHDLU)  
Register ONICHDLU is a use register. The scan rate is 10 s. Register ONICHDLU records if CAMA positions that use ONI are assigned to calls.

#### **Associated registers**

There are no associated registers.

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

There are no associated logs.

### **ONIDELGT**

#### **Register type**

Peg

#### **Description**

Operator number identification delegation (ONIDELGT)

Register ONIDELGT counts calls that wait in the CAMA call waiting queue for enough time for a register to increase.

Field MAZQ\_BEFORE\_OM in table CPOSTIME contains the maximum amount of time a call can wait in queue before a register increases.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIFDISC****Register type**

Peg

**Description**

Operator number identification forced disconnect (ONIFDISC)

Register ONIFDISC increases when a CAMA operator disconnects a call from the CAMA position.

The system routes disconnected calls to disconnect time-out (DISC) treatment.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIMBU****Register type**

Peg

**Description**

Operator number identification manual busy use (OMIBU)

Register ONIMBU is a use register. The scan rate is 10 s. Register ONIMBU records if CAMA positions that use ONI are manual busy or

seized. Register ONIMBU records if CAMA positions that use RONI are manual busy or seized.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIMTCHC****Register type**

Peg

**Description**

Operator number identification match check (ONIMTCHC)

Register ONIMTCHC increases when a CAMA operator enters a called number. The CAMA operator enters the called number in place of the calling number.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIOCCU****Register type**

Peg

**Description**

Operator number identification occupied (ONIOCCU)

Register ONIOCCU is a use register. The scan rate is 10 s. Register ONIOCCU records if CAMA positions that use ONI are assigned to or are available to handle calls. Register ONIOCCU records if CAMA positions that use ONI are available to handle calls. Register ONIOCCU also records if CAMA positions that use RONI are assigned to or are available to handle calls. Register ONIOCCU records if CAMA positions that use RONI are available to handle calls.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIOVFL****Register type**

Peg

**Description**

Operator number identification overflow (ONIOVFL)

Register ONIOVFL increases when a call attempts to enter the CAMA call waiting queue.

The call attempts to enter the CAMA call waiting queue when CAMA position is not available.

**Associated registers**

Register [ONIQOVFL](#) counts CAMA calls that the system routes to no service circuit (NOSC) treatment.

[ONIOVFL](#) - [ONIQOVFL](#) = number of calls that enter the CAMA call waiting queue.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIQABAN****Register type**

Peg

**Description**

Operator number identification queue abandon (ONIQABAN)

Register ONIQABAN counts calls abandoned in the CAMA call waiting queue.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIQOCC****Register type**

Peg

**Description**

Operator number identification queue occupied

Register ONIQOCC is a use register. The scan rate is 10 s. Register ONIQOCC records if calls wait for assignment to CAMA positions that use ONI. Register ONIQOCC records if calls wait for assignment to CAMA positions that use RONI.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIQOVFL****Register type**

Peg

**Description**

Operator number identification queue overflow (ONIQOVFL)

Register ONIQOVFL counts CAMA calls that route to no service circuit (NOSC) treatment.

The system routes calls to NOSC treatment because the CAMA call waiting queue maximum length exceeds the limit. Field DEFLECT in table CAMACSW contains the maximum CAMA call waiting queue length.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIQTOUT****Register type**

Peg

**Description**

Operator number identification queue timed out (ONIQTOUT)

Register ONIQTOUT counts calls that wait in the CAMA. The system routes calls to a treatment after a specified time.

Field MAXQ\_BEFORE\_TRTMT in table CPOSTIME contains the time-out period. The system routes calls to NOSC treatment.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONISBU****Register type**

Peg

**Description**

Operator number identification system busy (ONISBU)

Register ONISBU is a use register. The scan rate is 10 s. Register ONISBU records if CAMA positions that use ONI are system busy or peripheral module busy. Register ONUSBUS records if CAMA positions that use RONI are system busy or peripheral module busy.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONISZRS****Register type**

Peg

**Description**

Operator number identification seizures (ONISZRS) Register  
ONISZRS counts calls that connect to a CAMA position. The operator at the position acknowledges the calls.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**ONIWGCA****Register type**

Peg

**Description**

Operator number identification wrong code added (ONIWGCA)

Register ONIWGCA increases when the system rejects the entry that the CAMA operator makes.

The CAMA operator can enter NXX codes, trouble codes, or digits. The DMS-100 system rejects the following:

- NXX codes that are not correct
- trouble codes that are not correct
- digits that the system does not recognize

When the system detects digits that are not known, the system begins automatic testing procedures.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



## OPCHOICE

### Description

OM group Operator Choice (OPCHOICE) count enables you to determine the number of calls using each OPCHIDX. The OPCHIDX OM contains 255 tuples (one tuple for each OPCHIDX, 1-255). There is no tuple for OCHIDX 0. All tuples in the OPCHOICE OM group are present at all times, regardless of whether OPCHIDX is datafilled in table OPCHOICE.

The following table lists the key and info fields associated with OM group OPCHOICE:

| Key field      | Info field |
|----------------|------------|
| OPCHOICE Index | None       |

### Related functional groups

There are no functional groups associated with OM group OPCHOICE.

### Registers

The following table lists the registers associated with OM group OPCHOICE and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group OPCHOICE

| Register name           | Measures              |
|-------------------------|-----------------------|
| <a href="#">OPCHRTE</a> | OPCHOICE calls routed |

#### OPCHRTE

##### Register type

Peg

##### Description

The switch increments register OPCHRTE (OPCHOICE calls routed) each time it routes a 0-, 0+, or 01+ call using the OPCHOICE index (OPCHIDX). OPCHRTE pegs according to OPCHIDX. For cases where route advancing occurs, the switch only pegs the OPCHRTE register once (OPCHRTE is the register for the OPCHOICE index that was used to route the call.)

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## OracleRepLinkStats

### Description

OracleRepLinkStats tracks the following replication information for the Oracle replication link:

- Deferred transactions: The number of transactions queued for propagation to the destination database instance.
- Link Errors: The number of errors at the local database instance caused by deferred transactions

The following table lists the key and info fields associated with OM group OracleRepLinkStats.

| Key field             | Description                                                                             |
|-----------------------|-----------------------------------------------------------------------------------------|
| Deferred transactions | The number of transactions queued for propagation to the destination database instance. |
| Link Errors           | The number of errors at the local database instance caused by deferred transactions     |

### Related functional groups

The following functional groups are related to OM group OracleRepLinkStats:

- Database

### Registers

The following table lists the registers associated with OM group OracleRepLinkStats and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OracleRepLinkStats

| Register name                                       | Measures                                 |
|-----------------------------------------------------|------------------------------------------|
| <a href="#">minLinkDeferredTransactionsValue</a>    | minimum link deferred transactions value |
| <a href="#">medianLinkDeferredTransactionsValue</a> | median link deferred transactions value  |
| <a href="#">maxLinkDeferredTransactionsValue</a>    | maximum link deferred transactions value |
| <a href="#">minLinkErrorsValue</a>                  | minimum link errors value                |

**Registers for OM group OracleRepLinkStats**

| Register name                         | Measures                  |
|---------------------------------------|---------------------------|
| <a href="#">medianLinkErrorsValue</a> | median link errors value  |
| <a href="#">maxLinkErrorsValue</a>    | maximum link errors value |

**minLinkDeferredTransactionsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the minimum sampled value of the number of deferred transactions for the replication link.

**Associated registers**

[medianLinkDeferredTransactionsValue](#),  
[maxLinkDeferredTransactionsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 826

**medianLinkDeferredTransactionsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the median sampled value of the number of deferred transactions for the replication link.

**Associated registers**

[minLinkDeferredTransactionsValue](#),  
[maxLinkDeferredTransactionsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 826

**maxLinkDeferredTransactionsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the maximum sampled value of the number of deferred transactions for the replication link.

**Associated registers**

[minLinkDeferredTransactionsValue](#),  
[medianLinkDeferredTransactionsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 826

**minLinkErrorsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the minimum sampled value of the link errors for the replication link.

**Associated registers**

[medianLinkErrorsValue](#), [maxLinkErrorsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 727

**medianLinkErrorsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the median sampled value of the link errors for the replication link.

**Associated registers**

[minLinkErrorsValue](#), [maxLinkErrorsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 727

**maxLinkErrorsValue****Register type**

Usage

**Scan rate****Description**

In a given OM collection period, this register tracks the maximum sampled value of the link errors for the replication link.

**Associated registers**

[minLinkErrorsValue](#), [medianLinkErrorsValue](#)

**Extension registers**

None

**Associated logs**

DBMN 727

**Thresholding and Alarms**

The DBMN 826 alarm is associated with the deferred transaction register of this OM group. This alarm is raised when a backlog of deferred transactions is detected. The alarm is automatically cleared when the deferred transactions back log clears.

The DBMN 727 alarm is associated with the link error register of this OM group. This alarm is raised when the sampled number of link errors found is greater than 0. The alarm is automatically cleared when the number of link errors found is 0.

## OracleSysTableCounters

### Description

The OracleSysTableCounters OM Group captures Oracle specific information.

The following table lists the key fields associated with OM groupOracleSysTableCounters.

| Key field       | Description                                                                                                                                                                                                    |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Block Gets      | Oracle specific performance counter for block access obtained from the SNMP Oracle Database MIB variable "oraDbSysDbBlockGets."                                                                                |
| Consistent Gets | Oracle specific performance counter for block access obtained from the SNMP Oracle Database MIB variable "oraDbSysDbConsistentGets."                                                                           |
| Physical Reads  | Oracle specific performance counter that is incremented each time a block is retrieved from the disk subsystem. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbPhysReads." |
| Physical Writes | Oracle specific performance counter that is incremented each time a block is written to disk. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbPhysWrites."                  |
| Redo Entries    | Oracle specific performance counter that is incremented each time REDO data is copied into the log buffer. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbRedoEntries."    |



| Key field               | Description                                                                                                                                                                                                                                |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Redo Log Space Requests | Oracle specific performance counter that tracks the number of waits encountered when attempting to acquire an entry in the REDO log. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbLogSpaceRequests." |
| Disk Sorts              | Oracle specific performance counter that is incremented for each sort that results in disk space allocation. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbSortsDisk."                                |
| User Calls              | Oracle specific performance counter that tracks parse, execution and fetch operations. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbUserCalls."                                                      |
| User Commits            | Oracle specific performance counter that tracks the number of transactions that have successfully completed. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbUserCommits."                              |
| User Rollbacks          | Oracle specific performance counter tracks the number of aborted transactions. This information is obtained from the SNMP Oracle Database MIB variable "oraDbSysDbUserRollbacks."                                                          |

## Related functional groups

The following functional groups are related to OM group OracleSysTableCounters:

- Database

## Registers

The following table lists the registers associated with OM group OracleSysTableCounters and what they measure. For a description of a register, click on the register name.

### Registers for OM group OracleSysTableCounters

| Register name                        | Measures                |
|--------------------------------------|-------------------------|
| <a href="#">dbBlockGets</a>          | database block gets     |
| <a href="#">physReads</a>            | physical reads          |
| <a href="#">physWrites</a>           | physical writes         |
| <a href="#">redoEntries</a>          | redo entries            |
| <a href="#">redoLogSpaceRequests</a> | redo log space requests |
| <a href="#">sortsDisk</a>            | sorts disk              |
| <a href="#">userCalls</a>            | user calls              |
| <a href="#">userCommits</a>          | user commits            |
| <a href="#">userRollbacks</a>        | user rollbacks          |
| <a href="#">consistentGets</a>       | consistent gets         |

#### dbBlockGets

##### Register type

Peg

##### Description

In a given OM collection period, this register tracks the number of block gets.

##### Associated registers

None

##### Extension registers

None

**Associated logs**

None

**physReads****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of physical reads.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**physWrites****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of physical writes.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**redoEntries****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of redo entries.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**redoLogSpaceRequests****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of redo log space requests.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**sortsDisk****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of sorts that require disk space allocation.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**userCalls****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of user calls.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**userCommits****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of committed transactions.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**userRollbacks****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of aborted transactions.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**consistentGets****Register type**

Peg

**Description**

In a given OM collection period, this register tracks the number of consistent gets.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## OSACCP1

### Description

#### OSAC Call Processing 1

This OM group pegs a register for each Operator Services Systems Advanced Intelligent Network Centralization (OSAC) call processing operation and response on a per session pool basis. This group is pegged each time the switch sends or receives an OSAC call processing message.

OM group OSACCP1 provides up to 4095 tuples per office. The following table lists the key and info fields associated with OM group OSACCP1.

| Key field                                                                                                                                                                                                                                                                                                                                                                   | Info field                                                                          |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <p>OSACCP1 can be indexed by either of the following:</p> <ul style="list-style-type: none"><li>• SESNPLID {0 to 4094}: Key field for OASESNPL.</li><li>• SESNPLNM: Name associated with SESNPLID.</li></ul> <p>Only session pools defined as OSAC session pools can be indexed for OSACCP1.</p> <p>Datafilled in table OASESNPL with Orig Type = OSACORIG or OSACTERM.</p> | <p>OSAC_SP_INDEX_REGISTERI<br/>NFO - This name can be up to 16 characters long.</p> |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSACCP1.

## Registers

The following table lists the registers associated with OM group OSACCP1 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OSACCP1

| Register name            | Measures                                                  |
|--------------------------|-----------------------------------------------------------|
| <a href="#">ENDSESN</a>  | Register End Session Inform                               |
| <a href="#">GOTSESN</a>  | Register Got Session Inform                               |
| <a href="#">RELSENE</a>  | Register Release Session Error Response                   |
| <a href="#">RELSSENS</a> | Register Release Session Success Response                 |
| <a href="#">RELSESN</a>  | Register Release Session Request                          |
| <a href="#">RELSNI</a>   | Register Release Session Inform                           |
| <a href="#">SESBREQ</a>  | Register Session Begin Request                            |
| <a href="#">SESBREQE</a> | Register Session Begin Request Error Response             |
| <a href="#">SESBREQS</a> | Register Session Begin Request Success Response           |
| <a href="#">SIVERE</a>   | Register Session Initiation Verification Error Response   |
| <a href="#">SIVERRQ</a>  | Register Session Initiation Verification Request          |
| <a href="#">SIVERS</a>   | Register Session Initiation Verification Success Response |

#### ENDSESN

##### Register type

Peg

##### Description

Register End Session Inform (ENDSESN)

Register ENDSESN is pegged each time the corresponding operation or response is sent or received by the switch.

##### Associated registers

There are no associated registers.



**Extension registers**

ENDSESN2

**Associated logs**

There are no associated logs.

**GOTSESN****Register type**

Peg

**Description**

Register Got Session Inform (GOTSESN)

Register GOTSESN is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

GOTSESN2

**Associated logs**

There are no associated logs.

**RELSENE****Register type**

Peg

**Description**

Register Release Session Error Response (RELSENE)

Register RELSENE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

RELSENE2

**Associated logs**

There are no associated logs.

**RELSENS****Register type**

Peg

**Description**

Register Release Session Success Response (RELSENS) Register RELSENS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

RELSENS2

**Associated logs**

There are no associated logs.

**RELSESN****Register type**

Peg

**Description**

Register Release Session Request (RELSESN)

Register RELSESN is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RELSNI****Register type**

Peg

**Description**

Register Release Session Inform (RELSNI)

Register RELSNI is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

RELSNI2

**Associated logs**

There are no associated logs.

**SESBQR****Register type**

Peg

**Description**

Register Session Begin Request (SESBQR)

Register SESBQR is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SESBQR2

**Associated logs**

There are no associated logs.

**SESBRQE****Register type**

Peg

**Description**

Register Session Begin Request Error Response (SESBRQE)

Register SESBRQE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SESBRQE2

**Associated logs**

There are no associated logs.

**SESBRQS****Register type**

Peg

**Description**

Register Session Begin Request Success Response (SESBRQS)

Register SESBRQS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SESBRQS2

**Associated logs**

There are no associated logs.

**SIVERE****Register type**

Peg

**Description**

Register Session Initiation Verification Error Response (SIVERE)

Register SIVERE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SIVERE2

**Associated logs**

There are no associated logs.

**SIVERRQ****Register type**

Peg

**Description**

Register Session Initiation Verification Request (SIVERRQ)

Register SIVERRQ is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SIVERRQ2

**Associated logs**

There are no associated logs.

**SIVERS****Register type**

Peg

**Description**

Register Session Initiation Verification Success Response (SIVERS)

Register SIVERS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SIVERS2

**Associated logs**

There are no associated logs.

## OSACCP2

### Description

#### OSAC Call Processing 2

This OM group pegs a register for each Operator Services Systems Advanced Intelligent Network Centralization (OSAC) call processing operation and response on a per session pool basis. This group is pegged each time the switch sends or receives an OSAC call processing message.

OM group OSACCP2 provides up to 4095 tuples per office. A tuple is added to this OM group for each OSAC session pool defined in table OASESNPL.

The following table lists the key and info fields associated with OM group OSACCP2.

| Key field                                                                                                                                                                                                                                                                                                                          | Info field                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <p>OSACCP2 can be indexed by either of the following:</p> <ul style="list-style-type: none"><li>• SESNPLID {0 to 4094}: Key field for OASESNPL.</li><li>• SESNPLNM: Name associated with SESNPLID.</li></ul> <p>Only session pools defined in table OASESNPL with ORIGTYPE of OSACORIG or OSACTERM can be indexed for OSACCP2.</p> | <p>OSAC_SP_INDEX_REGISTERI<br/>NFO - This name can be up to 16 characters long.</p> |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSACCP2. ENSV is changed to OSAN in TOPS09.

## Registers

The following table lists the registers associated with OM group OSACCP2 and what they measure. For a description of a register, click on the register name.

### Registers for OM group OSACCP2

| Register name           | Measures                                    |
|-------------------------|---------------------------------------------|
| <a href="#">MISUPDT</a> | Management information system update inform |
| <a href="#">VCCONN</a>  | Register Voice Connect Request              |
| <a href="#">VCCONNE</a> | Register Voice Connect Error Response       |
| <a href="#">VCCONNS</a> | Register Voice Connect Success Response     |
| <a href="#">VCRELS</a>  | Register Voice Release Request              |
| <a href="#">VCRELSE</a> | Register Voice Release Error Response       |
| <a href="#">VCRELSS</a> | Register Voice Release Success Response     |

#### MISUPDT

**Register type**

Peg

**Description**

Management information system update inform (MISUPDT)

Register MISUPDT is pegged each time the MIS Update Inform operation is sent by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

MISUPDT2

**Associated logs**

There are no associated logs.

#### VCCONN

**Register type**

Peg

**Description**

Register Voice Connect Request (VCCONN)

Register VCCONN is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

VCCONN2

**Associated logs**

There are no associated logs.

**VCCONNE****Register type**

Peg

**Description**

Register Voice Connect Error Response (VCCONNE)

Register VCCONNE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**VCCONNS****Register type**

Peg

**Description**

Register Voice Connect Success Response (VCCONNS)

Register VCCONNS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.



**Extension registers**

VCCONNS2

**Associated logs**

There are no associated logs.

**VCRELS****Register type**

Peg

**Description**

Register Voice Release Request (VCRELS)

Register VCRELS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

VCRELS2

**Associated logs**

There are no associated logs.

**VCRELSE****Register type**

Peg

**Description**

Register Voice Release Error Response (VCRELSE) Register VCRELSE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

VCRELSE2

**Associated logs**

There are no associated logs.

**VCRELSS****Register type**

Peg

**Description**

Register Voice Release Success Response (VCRELSS) Register  
VCRELSS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

VCRELSS2

**Associated logs**

There are no associated logs.

## OSACND

### Description

#### OSAC Node Maintenance

This OM group pegs a register for each Operator Services Systems Advanced Intelligent Network Centralization (OSAC) Node Class operation and response on a per node basis. This group is pegged each time the switch sends or receives an OSAC Node Class message.

OM group OSACND provides up to 768 tuples per office. The following table lists the key and info fields associated with OM group OSACND.

| Key field                                                                                                                                                                                                                                                                                                                          | Info field                                                                            |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| <p>OSACND can be indexed by either of the following:</p> <ul style="list-style-type: none"><li>• NODEID {0 to 767}: Key field for table OANODINV.</li><li>• NODENAME: Name associated with NODEID.</li></ul> <p>Only nodes defined as OSAC nodes can be indexed for OSACND. Datafilled in table OANODINV with PM Type of OSAC.</p> | <p>OSAC_NODE_INDEX_REGIST<br/>ERINFO - This name can be up to 16 characters long.</p> |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSACND.

### Registers

The following table lists the registers associated with OM group OSACND and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OSACND

| Register name          | Measures                           |
|------------------------|------------------------------------|
| <a href="#">NDAUD</a>  | Register Node Audit Request        |
| <a href="#">NDAUDE</a> | Register Node Audit Error Response |

**Registers for OM group OSACND**

| Register name          | Measures                             |
|------------------------|--------------------------------------|
| <a href="#">NDAUDS</a> | Register Node Audit Success Response |
| <a href="#">NDBSY</a>  | Register Node Busy Request           |
| <a href="#">NDBSYE</a> | Register Node Busy Error Response    |
| <a href="#">NDBSYS</a> | Register Node Busy Success Response  |
| <a href="#">NDRTS</a>  | Register Node RTS Request            |
| <a href="#">NDRTSE</a> | Register Node RTS Error Response     |
| <a href="#">NDRTSS</a> | Register Node RTS Success Response   |
| <a href="#">NDTST</a>  | Register Node Test Request           |
| <a href="#">NDTSTE</a> | Register Node Test Error Response    |
| <a href="#">NDTSTS</a> | Register Node Test Success Response  |

**NDAUD****Register type**

Peg

**Description**

Register Node Audit Request (NDAUD)

Register NDAUD is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDAUD2

**Associated logs**

There are no associated logs.

**NDAUDE****Register type**

Peg

**Description**

Register Node Audit Error Response (NDAUDE)

Register NDAUDE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDAUDE2

**Associated logs**

There are no associated logs.

**NDAUDS****Register type**

Peg

**Description**

Register Node Audit Success Response (NDAUDS)

Register NDAUDS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDAUDS2

**Associated logs**

There are no associated logs.

**NDBSY****Register type**

Peg

**Description**

Register Node Busy Request (NDBSY)

Register NDBSY is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDBSY2

**Associated logs**

There are no associated logs.

**NDBSYE****Register type**

Peg

**Description**

Register Node Busy Error Response (NDBSYE)

Register NDBSYE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDBSYE2

**Associated logs**

There are no associated logs.

**NDBSYS****Register type**

Peg

**Description**

Register Node Busy Success Response (NDBSYS)

Register NDBSYS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDBSYS2

**Associated logs**

There are no associated logs.

**NDRTS****Register type**

Peg

**Description**

Register Node RTS Request (NDRTS)

Register NDRTS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDRTS2

**Associated logs**

There are no associated logs.

**NDRTSE****Register type**

Peg

**Description**

Register Node RTS Error Response (NDRTSE)

Register NDRTSE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDRTSE2

**Associated logs**

There are no associated logs.

**NDRTSS****Register type**

Peg

**Description**

Register Node RTS Success Response (NDRTSS)

Register NDRTSS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDRTSS2

**Associated logs**

There are no associated logs.

**NDTST****Register type**

Peg

**Description**

Register Node Test Request (NDTST)

Register NDTST is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**NDTSTE****Register type**

Peg

**Description**

Register Node Test Error Response (NDTSTE)

Register NDTSTE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDTSTE2



**Associated logs**

There are no associated logs.

**NDTSTS****Register type**

Peg

**Description**

Register Node Test Success Response (NDTSTS)

Register NDTSTS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

NDTSTS2

**Associated logs**

There are no associated logs.

## OSACSP

### Description

#### OSAC Session Pool Maintenance

This OM group pegs a register for each Operator Services Systems Advanced Intelligent Network Centralization (OSAC) Session Pool operation and response on a per session pool basis. This group is pegged each time the switch sends or receives an OSAC Session Pool Class message.

OM group OSACSP provides up to 4095 tuples per office. The following table lists the key and info fields associated with OM group OSACSP.

| Key field                                                                                                                                                                                                                                                                                                                                                                    | Info field                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <p>OSACSP can be indexed by either one of the following:</p> <ul style="list-style-type: none"><li>• SESNPLID {0 to 4094}: Key field for table OASESNPL.</li><li>• SESNPLNM: Name associated with SESNPLID.</li></ul> <p>Only session pools defined as OSAC session pools can be indexed for OSACSP. Datafilled in table OASESNPL with Orig Type = OSACORIG or OSACTERM.</p> | <p>OSAC_SP_INDEX_REGISTERI<br/>NFO - This name can be up to 16 characters long.</p> |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSACSP.

## Registers

The following table lists the registers associated with OM group OSACSP and what they measure. For a description of a register, click on the register name.

### Registers for OM group OSACSP

| Register name           | Measures                                     |
|-------------------------|----------------------------------------------|
| <a href="#">SPLAUD</a>  | Register Session Pool Audit Request          |
| <a href="#">SPLAUDE</a> | Register Session Pool Audit Error Response   |
| <a href="#">SPLAUDS</a> | Register Session Pool Audit Success Response |
| <a href="#">SPLBSY</a>  | Register Session Pool Busy Request           |
| <a href="#">SPLBSYE</a> | Register Session Pool Busy Error Response    |
| <a href="#">SPLBSYS</a> | Register Session Pool Busy Success Response  |
| <a href="#">SPLDRN</a>  | Register Session Pool Drain Request          |
| <a href="#">SPLRTS</a>  | Register Session Pool RTS Request            |
| <a href="#">SPLRTSE</a> | Register Session Pool RTS Error Response     |
| <a href="#">SPLRTSS</a> | Register Session Pool RTS Success Response   |
| <a href="#">SPLTST</a>  | Register Session Pool Test Request           |
| <a href="#">SPLTSTE</a> | Register Session Pool Test Error Response    |
| <a href="#">SPLTSTS</a> | Register Session Pool Test Success Response  |

### SPLAUD

#### Register type

Peg

#### Description

Register Session Pool Audit Request (SPLAUD)

Register SPLAUD is pegged each time the corresponding operation or response is sent or received by the switch.

#### Associated registers

There are no associated registers.

**Extension registers**

SPLAUD2

**Associated logs**

There are no associated logs.

**SPLAUDE****Register type**

Peg

**Description**

Register Session Pool Audit Error Response (SPLAUDE)

Register SPLAUDE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLAUDE2

**Associated logs**

There are no associated logs.

**SPLAUDS****Register type**

Peg

**Description**

Register Session Pool Audit Success Response (SPLAUDS)

Register SPLAUDS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLAUDS2

**Associated logs**

There are no associated logs.

**SPLBSY****Register type**

Peg

**Description**

Register Session Pool Busy Request (SPLBSY)

Register SPLBSY is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLBSY2

**Associated logs**

There are no associated logs.

**SPLBSYE****Register type**

Peg

**Description**

Register Session Pool Busy Error Response (SPLBSYE)

Register SPLBSYE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLBSYE2

**Associated logs**

There are no associated logs.

**SPLBSYS****Register type**

Peg

**Description**

Register Session Pool Busy Success Response (SPLBSYS)

Register SPLBSYS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLBSYS2

**Associated logs**

There are no associated logs.

**SPLDRN****Register type**

Peg

**Description**

Register Session Pool Drain Request (SPLDRN)

Register SPLDRN is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLDRN2

**Associated logs**

There are no associated logs.

**SPLRTS****Register type**

Peg

**Description**

Register Session Pool RTS Request (SPLRTS)

Register SPLRTS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLRTS2

**Associated logs**

There are no associated logs.

**SPLRTSE****Register type**

Peg

**Description**

Register Session Pool RTS Error Response (SPLRTSE)

Register SPLRTSE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLRTSE2

**Associated logs**

There are no associated logs.

**SPLRTSS****Register type**

Peg

**Description**

Register Session Pool RTS Success Response (SPLRTSS)

Register SPLRTSS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLRTSS2

**Associated logs**

There are no associated logs.

**SPLTST****Register type**

Peg

**Description**

Register Session Pool Test Request (SPLTST)

Register SPLTST is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLTST2

**Associated logs**

There are no associated logs.

**SPLTSTE****Register type**

Peg

**Description**

Register Session Pool Test Error Response (SPLTSTE)

Register SPLTSTE is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

SPLTSTE2

**Associated logs**

There are no associated logs.

**SPLTSTS****Register type**

Peg

**Description**

Register Session Pool Test Success Response (SPLTSTS)

Register SPLTSTS is pegged each time the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.



**Extension registers**

SPLTSTS2

**Associated logs**

There are no associated logs.

## OSNND

### Description

#### OSN Node

This OM group pegs a register for each Operator Services Node (OSN) Node Class operation and response on a per node basis. This group is pegged each time the switch sends or receives an OSN Node Class message.

OM group OSNND provides up to 768 tuples per office. The following table lists the key and info fields associated with OM group OSNND.

| Key field                                                                                                                                                                                                                                               | Info field                                                                    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| OSNND can be indexed by either of the following: <ul style="list-style-type: none"><li>• NODEID {0 to 767}: Key field for table OANODINV.</li><li>• NODENAME: Name associated with NODEID.</li></ul> Datafilled in table OANODINV with PM Type of OSAC. | OSAC_NODE_INDEX_REGIST<br>ERINFO - This name can be up to 16 characters long. |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSNND.

### Registers

The following table lists the registers associated with OM group OSNND and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OSNND

| Register name           | Measures                               |
|-------------------------|----------------------------------------|
| <a href="#">ONDBSYI</a> | OSN Node Busy Inform                   |
| <a href="#">ONDDFL</a>  | OSN Node Datafill Check Request        |
| <a href="#">ONDDFLE</a> | OSN Node Datafill Check Error Response |

**Registers for OM group OSNND**

| Register name           | Measures                                 |
|-------------------------|------------------------------------------|
| <a href="#">ONDDFLS</a> | OSN Node Datafill Check Success Response |
| <a href="#">ONDRTSI</a> | OSN Node RTS Inform                      |

**ONDBSYI****Register type**

Peg

**Description**

Register OSN Node Busy Inform (ONDBSYI)

Register ONDBSYI is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

ONDBSYI2

**Associated logs**

There are no associated logs.

**ONDDFL****Register type**

Peg

**Description**

Register OSN Node Datafill Check Request (ONDDFL)

Register ONDDFL is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

ONDDFL2

**Associated logs**

There are no associated logs.

**ONDDFLE****Register type**

Peg

**Description**

Register OSN Node Datafill Check Error Response (ONDDFLE)

Register ONDDFLE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

ONDDFLE2

**Associated logs**

There are no associated logs.

**ONDDFLS****Register type**

Peg

**Description**

Register OSN Node Datafill Check Success Response (ONDDFLS)

Register ONDDFLS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

ONDDFLS2

**Associated logs**

There are no associated logs.

**ONDRTSI****Register type**

Peg

**Description**

Register OSN Node RTS Inform (ONDRTSI)

Register ONDRTSI is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

ONDRTSI2

**Associated logs**

There are no associated logs.

## OSNSP

### Description

#### OSN Session Pool

This OM group pegs a register for each Operator Services Node (OSN) Session Pool Class operation and response on a per node session pool basis. This group is pegged each time the switch sends or receives an OSN Session Pool Class message.

OM group OSNSP provides up to 4095 tuples per office. The following table lists the key and info fields associated with OM group OSNSP.

| Key field                                                                                                                                                                                                                                               | Info field                                                                    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| OSNSP can be indexed by either of the following: <ul style="list-style-type: none"><li>• NODEID {0 to 767}: Key field for table OANODINV.</li><li>• NODENAME: Name associated with NODEID.</li></ul> Datafilled in table OANODINV with PM Type of OSAC. | OSAC_NODE_INDEX_REGIST<br>ERINFO - This name can be up to 16 characters long. |

### Related functional groups

Functional group Enhanced Services (ENSV0001) is associated with OM group OSNSP.

### Registers

The following table lists the registers associated with OM group OSNSP and what they measure. For a description of a register, click on the register name.

#### Registers for OM group OSNSP

| Register name           | Measures                                       |
|-------------------------|------------------------------------------------|
| <a href="#">OSPBSYI</a> | OSN Session Pool Busy Inform                   |
| <a href="#">OSPDFL</a>  | OSN Session Pool Datafill Check Request        |
| <a href="#">OSPDFLE</a> | OSN Session Pool Datafill Check Error Response |

**Registers for OM group OSNSP**

| Register name           | Measures                                         |
|-------------------------|--------------------------------------------------|
| <a href="#">OSPDFLS</a> | OSN Session Pool Datafill Check Success Response |
| <a href="#">OSPRTSI</a> | OSN Session Pool RTS Inform                      |

**OSPBSYI****Register type**

Peg

**Description**

Register OSN Session Pool Busy Inform (OSPBSYI)

Register OSPBSYI is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

OSPBSYI2

**Associated logs**

There are no associated logs.

**OSPDFL****Register type**

Peg

**Description**

Register OSN Session Pool Datafill Check Request (OSPDFL)

Register OSPDFL is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

OSPDFL2

**Associated logs**

There are no associated logs.

**OSPDFLE****Register type**

Peg

**Description**

Register OSN Session Pool Datafill Check Error Response (OSPDFLE)

Register OSPDFLE is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

OSPDFLE2

**Associated logs**

There are no associated logs.

**OSPDFLS****Register type**

Peg

**Description**

Register OSN Session Pool Datafill Check Success Response (OSPDFLS)

Register OSPDFLS is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

OSPDFLS2

**Associated logs**

There are no associated logs.

**OSPRTSI****Register type**

Peg

**Description**

Register OSN Session Pool RTS Inform (OSPRTSI)



Register OSPRTSI is pegged when the corresponding operation or response is sent or received by the switch.

**Associated registers**

There are no associated registers.

**Extension registers**

OSPRTSI2

**Associated logs**

There are no associated logs.

---

## OTS

---

### Description

OM group Office Traffic Summary (OTS) counts calls by source and destination. Sources can be trunk, line, or system generated. Each register in OTS is divided into three categories: originating traffic, incoming traffic, and system-generated traffic.

Originating traffic registers are: [NORG](#), [ORGTRM](#), [ORGOUT](#), [ORGTRMT](#), [ORGABDN](#), [ORGLKT](#) and [ORGFSET](#). Registers ORGTRMT and ORGFSET count requests for off-board service updates or changes. With the Off-board Service Control feature, off-board service updates peg the registers when subscribers do one of the following from an off-board application:

- query the status and/or programmed information of corresponding subscribed-to services
- generate a request to activate, deactivate or change the services

For more information about the Off-board Service Control feature, refer to *Advanced Intelligent Network Essentials Service Implementation Guide*, 297-5161-021, and the *Advanced Intelligent Network Essentials Service Enablers*, 297-5161-022.

*The Off-board Service Control feature applies only to DMS.*

Incoming traffic registers are: [NINC](#), [INCTRM](#), [INCOUT](#), [INCTRMT](#), [INCABNM](#), [INCABNC](#), [INCLKT](#) and [INCFSET](#).

System-generated traffic registers are: [NSYS](#), [SYSTRM](#), [SYSOUT](#), [SYSTRMT](#), [SYSABDN](#), [SYSLKT](#) and [SYSFSET](#).

OTS indicates the traffic load on the switch. The sum of the incoming calls represents the external traffic load on the switch. The sum of the originating and system-generated calls represents the internal traffic load on the switch. The sum of the internal and external traffic load is the total traffic load. The group also indicates the quality of service that the switch provides.

OTS contains information on the following types of calls offered to a DMS-100, DMS-200, or combined DMS-100/200 plain ordinary telephone service (POTS) office, except for common channel interoffice signaling, circuit-switched digital data service, and equal access:

- originating
- incoming
- system generated
- terminating
- outgoing
- other

Originating calls consist of line origination attempts.

Incoming calls consist of incoming attempts, including trunk, local test desk, remote office test line (ROTL), local collocated (operator) switchboard, toll collocated (operator) switchboard, and trunk test line (TTL) calls.

System-generated calls consist of call attempts that the system generates internally by the switch. System-generated calls include progressions or continuations of originating or incoming traffic, and calls that are not subscriber generated. For example, calls established by the alarm sending system or the service analysis DIALBACK system are not subscriber generated.

Terminating calls are calls that end on lines in the office.

Outgoing calls terminate on the following:

- outgoing trunks
- foreign potential test
- local test desks
- silent switchman
- centralized automatic message accounting (CAMA) positions
- test lines
- station ringer
- alarm checking feature

For terminating or outgoing traffic, registers increase when the network connection of the call is complete. When the terminating party is busy, the network connection does not complete. The system applies a tone and the line or trunk termination register increases.

The following calls do not use network module connections, but for OTS these calls are traffic calls:

- revertive calls
- service analysis dialback to line
- alarm sending to trunk
- line to station ringer
- line to silent switchman

Other traffic consists of calls that connect to tones or announcements (caused by error conditions), activations and deactivations of custom calling features, and abandoned or locked-out calls.

The following table lists the key and info fields associated with OM group OTS.

| Key field | Info field |
|-----------|------------|
| None      | None       |

The office parameter OFFICETYPE in table OFCSTD specifies the type of office. The value of OFFICETYPE controls the output of the office traffic summary group (OTS). The correct entries for OFFICETYPE are:

- OFF100
- OFF200
- OFFCOMB
- OFFCOMBLWW
- OFFCOMBITOPS

All registers are output in offices whose OFFICETYPE is OFF100, OFFCOMB, OFFCOMBLWW, and OFFCOMBITOPS.

The following registers are output in offices whose OFFICETYPE is OFF200: [NINC](#), [INCOUT](#), [INCTRMT](#), [INCABNM](#), [INCABNC](#), [INCLKT](#), [INCFSET](#), [NSYS](#), [SYSOUT](#), [SYSTRMT](#), [SYSABDN](#), [SYSLKT](#) and [SYSFSET](#).

## Related functional groups

The following functional groups are associated with functional groups for OM group OTS:

- DMS-100 Local
- DMS-200 Toll
- DMS 100/200 Combined local/toll

## Registers

The following table lists the registers associated with OM group OTS and what they measure. For a detailed description of a register, click on the register name.

### Registers for OM group OTS (Sheet 1 of 2)

| Register name           | Measures                            |
|-------------------------|-------------------------------------|
| <a href="#">INCABNC</a> | Incoming abandoned by the customer  |
| <a href="#">INCABNM</a> | Incoming abandoned by the machine   |
| <a href="#">INCFSET</a> | Incoming to custom calling features |
| <a href="#">INCLKT</a>  | Incoming to lockout                 |
| <a href="#">INCOUT</a>  | Incoming to outgoing connections    |
| <a href="#">INCTRM</a>  | Incoming to terminating connections |
| <a href="#">INCTRMT</a> | Incoming to a treatment             |
| <a href="#">NDCACT</a>  | NDC activation                      |
| <a href="#">NDCDACT</a> | NDC deactivation                    |
| <a href="#">NDCINTG</a> | NDC interrogation                   |
| <a href="#">NDCCEER</a> | NDC errors                          |
| <a href="#">NDCUSGE</a> | NDC usage                           |
| <a href="#">NDCFSET</a> | Originating feature set             |
| <a href="#">NINC</a>    | Incoming attempts                   |
| <a href="#">NORG</a>    | Originating attempts                |
| <a href="#">NSYS</a>    | System origination                  |

**Registers for OM group OTS (Sheet 2 of 2)**

| Register name           | Measures                                       |
|-------------------------|------------------------------------------------|
| <a href="#">ORGABDN</a> | Originating, abandoned                         |
| <a href="#">ORGFSET</a> | Originating to custom calling feature          |
| <a href="#">ORGLKT</a>  | Originating to lockout                         |
| <a href="#">ORGOUT</a>  | Originating to outgoing connections            |
| <a href="#">ORGTRM</a>  | Originating to terminating connections         |
| <a href="#">ORGTRMT</a> | Originating to treatment connections           |
| <a href="#">SYSABDN</a> | System originations abandoned                  |
| <a href="#">SYSFSET</a> | System originations to custom calling features |
| <a href="#">SYSLKT</a>  | System originations to lockout                 |
| <a href="#">SYSOUT</a>  | System to outgoing connections                 |
| <a href="#">SYSTRM</a>  | System to terminating connections              |
| <a href="#">SYSTRMT</a> | System to treatment                            |

**INCABNC****Register type**

Peg

**Description**

INCABNC counts incoming call attempts abandoned by the customer (subscriber). The calls are abandoned before they connect to a terminating line, outgoing trunk, tone, announcement, lockout status, or feature activation or deactivation.

**Associated registers**

The following registers are associated with INCABNC:

- [INCABNM](#)
- TRK\_PRERTEAB counts incoming calls abandoned by the machine or the subscriber. Calls are counted by trunk group.

**Note:** This relationship does not apply to calls that originate from a mobile telephone exchange (MTX).

- OFZ\_INABNC counts incoming calls abandoned by the subscriber before being processed.

**Validation formulas**

The following formulas relate to INCABNC and its associated registers:

- $\Sigma \text{TRK\_PRERTEAB} = \text{OTS\_INCABNM} + \text{OTS\_INCABNC}$
- $\text{OTS\_INCABNC} = \text{OFZ\_INABNC}$

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK162

**INCABNM****Register type**

Peg

**Description**

INCABNM counts incoming call attempts abandoned by the machine. The calls are abandoned before they connect to terminating traffic, outgoing traffic, a tone, an announcement, lockout status, or feature activation or deactivation.

**Associated registers**

The following registers are associated with INCABNM:

- [INCABNC](#)
- TRK\_PRERTEAB counts incoming calls abandoned by the machine or the subscriber. Calls are counted by trunk group.

**Note:** This relationship does not apply to calls that originate from a mobile telephone exchange (MTX).

- OFZ\_INABNM counts incoming calls that come in from a trunk and are abandoned by the machine before they are processed.

**Validation formulas**

The following formulas relate to INCABNM and its associated registers:

- $\Sigma \text{TRK\_PRERTEAB} = \text{OTS\_INCABNM} + \text{OTS\_INCABNC}$
- $\text{OTS\_INCABNM} = \text{OFZ\_INABNC}$

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK162

**INCFSET****Register type**

Peg

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**INCLKT****Register type**

Peg

**Description**

INCLKT counts incoming call attempts that fail to connect or receive a treatment. The call routes to lockout.

**Associated registers**

OFZ\_INLKT counts incoming calls that fail and route to lockout.

**Validation formula**

$\text{OTS\_INCLKT} - (\text{number of calls that fail caused to remote-end lockout}) = \text{OFZ\_INLKT}$

**Extension registers**

None

**Associated logs**

TRK111, TRK113, TRK122, TRK123



**INCOUT****Register type**

Peg

**Description**

INCOUT counts incoming call attempts that connect to an outgoing trunk.

**Associated registers**

None

**Extension registers**

INCOUT2

**Associated logs**

None

**INCTRM****Register type**

Peg

**Description**

INCTRM counts incoming call attempts that terminate to a line. The connection of a busy tone when a line is busy is a line termination. Register INCTRM counts line terminations.

**Associated registers**

None

**Extension registers**

INCTRM2

**Associated logs**

None

**INCTRMT****Register type**

Peg

**Description**

INCTRMT counts incoming call attempts that route to a tone or an announcement because of an error condition.

**Associated registers**

The following registers are associated with INCTRMT:

- [ORGTRMT](#)
- [SYSTRMT](#)
- ANN\_ANNATT counts calls that route to announcements.
- TONES\_TONEATT counts calls that route to tones.

**Validation formula**
$$\Sigma \text{ ANN\_ANNATT} + \Sigma \text{ TONES\_TONEATT} \text{ OTS\_INCTRMT} + \text{OTS\_ORGTRMT} + \text{OTS\_SYSTRMT}$$
**Extension registers**

None

**Associated logs**

TRK138

**NDCACT****Register type**

Peg

**Description**

NDCACT counts the number of times a subscriber activates INDC.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDCDACT****Register type**

Peg

**Description**

NDCDACT counts the number of times a subscriber deactivates INDC.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDCINTG****Register type**

Peg

**Description**

NDCINTG counts the number of times a subscriber interrogates the status of INDC.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDCCEER****Register type**

Peg

**Description**

NDCCEER counts the number of times a subscriber does not use INDC correctly. When a subscriber attempts to activate, deactivate or interrogate INDC without an assignment, the subscriber is not using INDC correctly.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDCUSGE****Register type**

Peg

**Description**

NDCUSGE counts the times a Call Waiting (CW) or Toll Break-in (TBI) call attempts to reach an INDC subscriber engaged in a call. The system prevents the interruption because INDC is active.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NDCFSET****Register type**

Peg

**Description**

NDCFSET counts the originating call attempts that activate or deactivate INDC.

**Associated registers**

NDCFSET2

**Extension registers**

None

**Associated logs**

None

**NINC****Register type**

Peg

**Description**

NINC counts incoming call attempts recognized by the central control. The intended destination of the call is a line, a trunk, an announcement, or a tone.

**Associated registers**

OTS\_NINC counts incoming calls. The following registers count each call according to its destination: [INCTRM](#), [INCOUT](#), [INCTRMT](#), [INCABNM](#), [INCABNC](#), [INCLKT](#), [INCFSET](#)

**Validation formula**
$$(65536 \times \text{NINC2}) + \text{NINC} = (65536 \times \text{INCTRM2}) + \text{INCTRM} + (65536 \times \text{INCOUT2}) + \text{INCOUT} + \text{INCTRMT} + \text{INCABNM} + \text{INCABNC} + \text{INCLKT} + \text{INCFSET}$$
**Extension registers**

NINC2

**Associated logs**

None

**NORG****Register type**

Peg

**Description**

NORG counts originating call attempts recognized by the central control. The intended destination of the call is a line, a trunk, an announcement, or a tone. Register NORG also counts originating call attempts that go immediately to lockout (caused by line load control). The system recognizes and counts a line involved in a call that flashes to attempt to initiate a three-way call.

**Associated registers**[ORGTRM](#), [ORGOUT](#), [ORGTRMT](#), [ORGABDN](#), [ORGLKT](#), [ORGFSET](#).**Validation formula**
$$(65536 \times \text{NORG2}) + \text{NORG} = (65536 \times \text{ORGTRM2}) + \text{ORGTRM} + (65536 \times \text{ORGOUT2}) + \text{ORGOUT} + \text{ORGTRMT} + \text{ORGABDN} + \text{ORGLKT} + (65536 \times \text{ORGFSET}) + \text{ORGFSET}$$
**Extension registers**

NORG2

**Associated logs**

None

**NSYS****Register type**

Peg

**Description**

NSYS counts calls that the central control (CC) recognizes as system-generated calls. System-generated calls include originations that are not included in NORG or NINC.

**Associated registers**

[SYSTRM](#), [SYSOUT](#), [SYSTRMT](#), [SYSABDN](#), [SYSLKT](#), [SYSFSET](#).

**Validation formula**

$(65536 \times \text{NSYS2}) + \text{NSYS} = \text{SYSTRM} + \text{SYSOUT} + \text{SYSTRMT} + \text{SYSABDN} + \text{SYSLKT} + \text{SYSFSET}$

**Extension registers**

NSYS2

**Associated logs**

None

**ORGABDN****Register type**

Peg

**Description**

ORGABDN counts originating call attempts that the subscriber abandons before they route to a terminating line, outgoing trunk, tone, announcement, lockout status, or feature activation or deactivation. If the line is on a line module, the system counts the flash of the switch hook that occurs when the caller dials.

**Associated registers**

OFZ\_ORIGABDN counts line originations of calls the caller abandons before the calls route to a trunk, line, or treatment.

**Validation formula**

OTS\_ORGABDN = OFZ\_ORIGABDN when both registers are pegged at the same time. The OTS group was created after the OFZ group and some older applications do not account for the newer OTS registers.

**Extension registers**

None

**Associated logs**

LINE106, LINE108, LINE138

**ORGFSET****Register type**

Peg

**Description**

ORGFSET counts originating call attempts that activate or deactivate a custom calling feature.

*For DMS only:* ORGFSET is pegged when a user successfully programs a Speed Call feature by using the Speed Call Update message (off-board service update).

**Associated registers**

None

**Extension registers**

ORGFETS2

**Associated logs**

None

**ORGLKT****Register type**

Peg

**Description**

ORGLKT counts originating call attempts that fail, route to lockout without connecting or going to treatment.

**Associated registers**

OFZ\_ORIGLKT counts originating call attempts that fail, route to lockout without connecting or going to treatment.

**Validation formula**

$OTS\_ORGLKT = OFZ\_ORIGLKT$

**Extension registers**

None

**Associated logs**

LINE104, LINE105, LINE109, LINE204, LINE138, NET130, OM2200

**ORGOUT****Register type**

Peg

**Description**

ORGOUT counts originating call attempts that connect to an outgoing trunk.

**Associated registers**

None

**Extension registers**  
ORGOUT2

**Associated logs**  
None

## **ORGTRM**

**Register type**  
Peg

**Description**  
ORGTRM counts originating call attempts that connect to terminating traffic and connection to busy tone terminations.

**Associated registers**  
None

**Extension registers**  
ORGTRM2

**Associated logs**  
None

## **ORGTRMT**

**Register type**  
Peg

**Description**  
ORGTRMT counts originating call attempts that connect to a tone or an announcement because of an error condition. The register also counts tones that are applied to indicate error conditions and are not determined to be a DMS treatment (for example, a three-way call activation error resulting in an error tone to a line).

*For DMS only:* ORGTRMT is pegged when the Speed Call Update (off-board service update) message fails to program the Speed Call feature because of an error condition.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
None



**SYSABDN****Register type**

Peg

**Description**

SYSABDN counts system-generated calls that are abandoned before they connect to a terminating line, outgoing trunk, tone, announcement, lockout status, or feature activation or deactivation.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

TRK114, TRK116, TRK162

**SYSFSET****Register type**

Peg

**Description**

SYSFSET counts system-generated calls that activate or deactivate a custom calling feature.

**Note:** This register does not increase in DMS-300 international offices.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SYSLKT****Register type**

Peg

**Description**

SYSLKT counts system-generated calls that fail to connect or receive a treatment and that route to lockout.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

TRK111, TRK113, TRK122, TRK123

**SYSOUT****Register type**

Peg

**Description**

SYSOUT counts system-generated calls that connect to an outgoing trunk.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SYSTRM****Register type**

Peg

**Description**

SYSTRM counts system-generated calls that terminate to a line, and connection to busy tone terminations.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SYSTRMT****Register type**

Peg

**Description**

SYSTRMT counts system-generated calls that route to a tone or an announcement because of an error condition.

**Associated registers**

The following registers are associated with SYSTRMT:

- [INCTRMT](#)
- [ORGTRMT](#)
- ANN\_ANNATT counts calls that route to announcements.
- TONES\_TONEATT counts calls that route to tones.

**Validation formula**
$$\Sigma \text{ ANN\_ANNATT} + \Sigma \text{ TONES\_TONEATT} \text{ OTS\_INCTRMT} + \text{OTS\_ORGTRMT} + \text{OTS\_SYSTRMT}$$
**Extension registers**

None

**Associated logs**

TRK138

---

## PCMCARR

---

### Description

Consultative Committee on International Telegraphy and Telephony (CCITT) DS30 digital carrier maintenance summary (PCMCARR)

The OM group PCMCARR provides information on pulse code modulated (PCM30) carriers. The PCM30 is a transmission standard that defines the characteristics of international digital trunks and transmission links.

The PCM30 trunks interface with international digital trunk controllers (IDTC). The PCM30 links provide voice and signaling channels between the very small remote (VSR) and the international line group controller (ILGC).

The OM group PCMCARR has 24 peg registers that count the following errors and faults:

- local loss of frame alignment (LLFA)
- local loss of multiframe alignment (LLMA)
- remote frame alarm indication (RFAI)
- remote multiframe alarm indication (RMAI)
- alarm indication signal (AIS)
- bit error rate (BER)
- frame slip (SLIP)
- signaling channel (SIGL)

The OM group PCMCARR has four usage registers that record the following PCM30 carrier states:

- system busy
- central side (C-side) busy
- peripheral side (P-side) busy
- manual busy

The OM group PCMCARR supplies the data to monitor the performance of PCM30 carriers.

The OM group PCMCARR provides two tuples for each PCM30 carrier. The following table lists the key and info fields associated with OM group PCMCARR.

| Key field | Info field              |
|-----------|-------------------------|
| none      | D30OMINF is a structure |

The D30OMINF structure contains the following information:

- SITE name of the program model (PM)
- PM name and external number
- D30 CIRCUIT number (0-31)
- CARRIER DIRECTION (C or P) that indicates if PM port is to the C-side or P-side of the carrier

Table CARRMTC defines PM maintenance data, out-of-service limits for alarms, maintenance limit for each D30 alarm type and system return-to-service information.

Field ACTION in table LTCPSINV specifies a PCM30 carrier is system busy when it reaches an out-of-service limit.

## Related functional groups

The associated functional groups for the OM group PCMCARR are:

- DMS-100 International
- DMS-300 International
- D30 carrier links

## Registers

The following table lists the registers associated with OM group PCMCARR and what they measure. For a description of a register, click on the register name.

### Registers for OM group PCMCARR

| Register name            | Measures                                          |
|--------------------------|---------------------------------------------------|
| <a href="#">AIS16ERR</a> | Alarm indication signal (AIS) in channel 16 error |
| <a href="#">AIS16FLT</a> | AIS in channel 16 fault                           |
| <a href="#">AISERR</a>   | AIS error                                         |

**Registers for OM group PCMCARR**

| Register name            | Measures                                                 |
|--------------------------|----------------------------------------------------------|
| <a href="#">AISFLT</a>   | AIS fault                                                |
| <a href="#">BERERR</a>   | Bit error rate (BER) error                               |
| <a href="#">BERFLT</a>   | BER fault                                                |
| <a href="#">CARRCBSY</a> | Carrier C-side busy usage                                |
| <a href="#">CARRMANB</a> | Carrier manual busy usage                                |
| <a href="#">CARRPBSY</a> | Carrier P-side busy usage                                |
| <a href="#">CARRSYSB</a> | Carrier system busy usage                                |
| <a href="#">CRC4ERR</a>  | Cyclic redundancy check 4 (CRC4) procedure error         |
| <a href="#">CRC4FLT</a>  | Cyclic redundancy check 4 (CRC4) procedure fault         |
| <a href="#">CREERR</a>   | Cyclic redundancy check 4 (CRC4) remote reporting enable |
| <a href="#">CREFLT</a>   | Cyclic redundancy check 4 (CRC4) remote reporting fault  |
| <a href="#">LLCMAERR</a> | Cyclic redundancy check 4 (CRC4) remote reporting fault  |
| <a href="#">LLCMAFLT</a> | Loss of local CRC4 multiframe alignment (LLCMA) fault    |
| <a href="#">LLFAERR</a>  | Local loss of frame alignment (LLFA) error               |
| <a href="#">LLFAFLT</a>  | Local loss of frame alignment (LLFA) fault               |
| <a href="#">LLMAERR</a>  | Local loss of multiframe alignment (LLMA) error          |
| <a href="#">LLMAFLT</a>  | Local loss of multiframe alignment (LLMA) fault          |
| <a href="#">RFAIERR</a>  | Remote frame alarm indication (RFAI) error               |
| <a href="#">RFAIFLT</a>  | Remote frame alarm indication (RFAI) fault               |
| <a href="#">RMAIERR</a>  | Remote multiframe alarm indication (RMAI) error          |

**Registers for OM group PCMCARR**

| Register name           | Measures                                        |
|-------------------------|-------------------------------------------------|
| <a href="#">RMAIFLT</a> | Remote multiframe alarm indication (RMAI) fault |
| <a href="#">SIGLERR</a> | Signaling channels error                        |
| <a href="#">SIGFLT</a>  | Signaling channels fault                        |
| <a href="#">SLIPERR</a> | Slip error                                      |
| <a href="#">SLIPFLT</a> | Slip fault                                      |

**AIS16ERR****Register type**

Peg

**Description**

Alarm indication signal (AIS) in channel 16 error (AIS16ERR)

Register AIS16ERR increases when the system detects an AIS16 error on the carrier.

**Associated registers**[AIS16FLT](#)**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**AIS16FLT****Register type**

Peg

**Description**

AIS in channel 16 fault (AIS16FLT)

Register AIS16FLT increases when an AIS16 error causes the D30 link to become busy. The error depends on the AIS16OST and AIS16OL limits set in table CARRMTC.

**Associated registers**

Register [AIS16ERR](#) increases when the carrier reports an AIS16 error when the threshold value is AIS16ML. The system raises the alarm when the maintenance limit (ML) threshold exceeds its maximum. The ML threshold data is in table CARRMTC.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**AISERR****Register type**

Peg

**Description**

AIS error (AISERR)

Register AISERR increases when a PCM30 carrier receives a continuous stream of ones (111...), which indicates an AIS error.

**Associated registers**

Register [AISFLT](#) counts AIS faults that make a PCM30 carrier system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**AISFLT****Register type**

Peg

**Description**

AIS fault (AISFLT)

Register AISFLT counts AIS faults that cause a PCM30 carrier to become system busy. AISFLT increases

- for each continuous AIS error
- when AIS errors that are not continuous reach the out-of-service limit (AISOL) and data are in table LTCPSINV to make the carrier system busy



A continuous AIS error persists long enough time to reach the out-of-service time limit (AISOST). The PM maintains an error count that is not continuous and resets it every 5 min.

**Associated registers**

Register [AISERR](#) increases when a PM30 carrier receives a continuous stream of ones (111...), which indicates an AIS error.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 either because the software executes improperly or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**BERERR****Register type**

Peg

**Description**

Bit error rate (BER) error (BERERR)

BERERR increases when the system detects a BER error on a PCM30 carrier.

**Associated registers**

Register [BERFLT](#) counts BER faults that make a PCM30 carrier system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**BERFLT****Register type**

Peg

**Description**

BER fault (BERFLT)

Register BERFLT counts BER faults that make a PCM30 carrier system busy.

Register BERFLT increases when the system detects BER errors on a PCM30 carrier. These errors reach the BER out-of-service limit (BEROL). The system enters data in table LTCPSINV to make the carrier system busy.

**Associated registers**

Register [BERERR](#) increases when the system detects a BER error in a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CARRCBSY****Register type**

Usage

**Description**

Carrier C-side busy usage (CARRCBSY)

Register CARRCBSY is a usage register. Every 100 s the system scans the PCM30 carriers. Register CARRCBSY records if a carrier is C-side busy because the C-side peripheral module (IDTC) is not in service.

**Associated registers**

Register [CARRSYSB](#) records if a PCM30 carrier is system busy because of a fault.

Register [CARRPBSY](#) records if a PCM30 carrier is P-side busy because the P-side peripheral (VSR) is not in service.

Register [CARRMANB](#) records if a PCM30 carrier is manual busy.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 it receives when carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software execution or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**CARRMANB****Register type**

Usage

**Description**

Carrier manual busy usage (CARRMANB)

Register CARRMANB is a usage register. Every 100 s the system scans the PCM30 carriers, and register CARRMANB records if a carrier is manually busy.

**Associated registers**

Register [CARRSYSB](#) records if a PCM30 carrier is system busy because of a fault.

Register [CARRCBSY](#) records if a PCM30 carrier is C-side busy because the C-side PM (IDTC) is not in service.

Register [CARRPBSY](#) records if a PCM30 carrier is P-side busy because the P-side peripheral VSR is not in service.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when it receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

## **CARRPBSY**

### **Register type**

Usage

### **Description**

Carrier P-side busy usage (CARRPBSY)

Register CARRPBSY is a usage register. Every 100 s the system scans the PCM30 carriers, and CARRPBSY records if a PCM30 carrier is P-side busy.

The PCM30 can be busy as a result of the P-side peripheral VSR not being in service.

### **Associated registers**

Register [CARRSYSB](#) records if a PCM30 carrier is system busy because of a fault.

Register [CARRCBSY](#) records if a PCM30 carrier is C-side busy because the C-side PM (IDTC) is not in service.

Register [CARRMANB](#) records if a PCM30 carrier is manual busy.

### **Extension registers**

There are no extension registers.

### **Associated logs**

The system generates PM110 when it receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 either because software executes improperly or because a hardware problem is affecting software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

## CARRSYSB

### Register type

Usage

### Description

Carrier system busy usage (CARRSYSB)

Register CARRSYSB is a usage register. Every 100 s the system scans the PCM30 carriers, and register CARRSYSB records if a carrier is system busy because of a fault.

### Associated registers

Register [CARRCBSY](#) records if a PCM30 carrier is C-side busy because the C-side PM (IDTC) is not in service.

Register [CARRPBSY](#) records if a PCM30 carrier is P-side busy because the P-side peripheral (VSR) is not in service.

Register [CARRMANB](#) records if a PCM30 carrier is manual busy.

### Extension registers

There are no extension registers.

### Associated logs

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

## CRC4ERR

### Register type

Peg

### Description

Cyclic redundancy check 4 (CRC4) procedure error (CRC4ERR)

Register CRC4ERR increases when the system detects a CRC4 error on the carrier.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CRC4FLT****Register type**

Peg

**Description**

Cyclic redundancy check 4 (CRC4) procedure fault (CRC4FLT)

Register CRC4FLT increases when a CRC4 error causes the link to become system busy. The CRC4OL and CRC4OST limits in table CARRMTC and the state of the set action boolean on CRC4OL define the error level.

**Associated registers**

Register [CRC4ERR](#) increases when the carrier reports a CRC4 error and the threshold value is CRC4ML.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM187 when a carrier link is system busy.

**CREERR****Register type**

Peg

**Description**

Cyclic redundancy check 4 (CRC4) remote reporting enable (CREERR)

Register CREERR counts the number CRC4 errors the system detects on the remote end where the threshold value is CRC4ML.

**Associated registers**

[CREFLT](#)

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CREFLT****Register type**

Peg

**Description**

Cyclic redundancy check 4 (CRC4) remote reporting fault (CREFLT)

Register CREFLT increases if a CRC4 error causes the link to become system busy. The CRC4 out-of-service limit (CRC4OL) in table CARRMTC defines the error level.

**Associated registers**

[CREERR](#)

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM187 when a carrier link becomes system busy.

**LLCMAERR****Register type**

Peg

**Description**

Loss of local CRC4 multiframe alignment (LLCMA) error (LLCMAERR)

Register LLCMAERR counts the number of times the system detects an LLCMA error on the carrier.

**Associated registers**

[LLCMAFLT](#)

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**LLCMAFLT****Register type**

Peg

**Description**

Loss of local CRC4 multiframe alignment (LLCMA) fault (LLCMAFLT)

Register LLCMAFLT increases when an LLCMA error causes the D30 link to become system busy. The CRC4 out-of-service time limit (CRC4OST) in table CARRMTC defines the error level.

**Associated registers**

[LLCMAERR](#)

**Extension registers**

There are no extension registers.

**Associated logs**

The PM subsystem generates PM187 when a carrier link becomes system busy.

**LLFAERR****Register type**

Peg

**Description**

Local loss of frame alignment (LLFA) error (LLFAERR)

Register LLFAERR increases when the system detects an error in three or four consecutive frame alignment patterns of a PCM30 carrier.

**Associated registers**

Register [LLFAFLT](#) counts frame alignment faults that make a PCM30 carrier system busy.

Register [LLMAERR](#) increases when the system detects an error in two consecutive multiframe alignment patterns of a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



**LLFAFLT****Register type**

Peg

**Description**

Local loss of frame alignment (LLFA) fault (LLFAFLT)

Register LLFAFLT counts frame alignment faults that cause a PCM30 carrier to become system busy. LLFAFLT increases:

- for each continuous LLFA error
- when LLFA errors that are not continuous reach the out-of-service limit (LLFAOL) and important data are in table LTCPSINV make the carrier system busy.

A continuous LLFA error is an error that persists long enough to reach the out-of-service time limit (LLFAOST). The error count for errors that are not continuous is in the peripheral module (PM). The error count resets every 5 min.

**Associated registers**

Register [LLFAERR](#) increases when the system detects an error in three or four consecutive frame alignment patterns of a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 either because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**LLMAERR****Register type**

Peg

**Description**

Local loss of multiframe alignment (LLMA) error (LLMAERR)

Register LLMAERR increases when the system detects an error in two consecutive multiframe alignment patterns in a PCM30 carrier.

**Associated registers**

Register [LLFAERR](#) increases when the system detects an error in three or four consecutive frame alignment patterns.

Register [LLMAFLT](#) counts multiframe alignment faults (LLMA) that make a PCM30 carrier system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**LLMAFLT****Register type**

Peg

**Description**

Local loss of multiframe alignment (LLMA) fault (LLMAFLT)

Register LLMAFLT counts LLMA faults that cause a PCM30 carrier to be system busy. LLMAFLT increases

- for each continuous LLMA error
- when LLMA errors that are not continuous reach the out-of-service limit (LLMAOL) and important data are in table LTCPSINV to allow the carrier to become system busy

A continuous LLMA error is an error that persists for enough time to reach the out-of-service time limit (LLMAOST). The error count for errors that are not continuous is in the peripheral module (PM). The error count is reset every 5 min.

**Associated registers**

Register [LLMAERR](#) increases when the system detects an error in two consecutive multiframe alignment patterns of a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**RFAIERR****Register type**

Peg

**Description**

Remote frame alarm indication (RFAI) error (RFAIERR)

Register RFAIERR increases when remote equipment reports a frame-level error, an equipment failure, or both in a PCM30 carrier.

**Associated registers**

Register [RFAIFLT](#) counts frame alarm indication faults that make a PCM30 carrier system busy.

Register [RMAIERR](#) increases when remote equipment reports a multiframe-level error, an equipment failure, or both in a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RFAIFLT****Register type**

Peg

**Description**

Remote frame alarm indication (RFAI) fault (RFAIFLT)

Register RFAIFLT counts RFAI faults that cause a PCM30 carrier to become system busy. RFAIFLT increases

- for each continuous RFAI error
- when RFAI errors that are not continuous reach the out-of-service limit (RFAIOL) and important data are in table LTCPSINV to allow the carrier to become system busy
- for remote PM equipment failures

A continuous RFAI error is an error that persists for enough time to reach the out-of-service time limit (RFAIOL). The error count for errors that are not continuous is in the peripheral module (PM). The error count is reset every 5 min.

**Associated registers**

Register PCMCARR\_ [RFAIERR](#) increases when remote equipment reports a frame-level error, an equipment failure, or both in a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 either because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**RMAIERR****Register type**

Peg

**Description**

Remote multiframe alarm indication (RMAI) error (RMAIERR)

Register RMAIERR increases when remote equipment reports a multiframe-level error, an equipment failure, or both in a PCM30 carrier.

**Associated registers**

Register [RFAIERR](#) increases when remote equipment reports a frame-level error, an equipment failure, or both in a PCM30 carrier.

Register RMAIFLT counts multiframe alarm indication faults that make a PCM30 carrier system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RMAIFLT****Register type**

Peg

**Description**

Remote multiframe alarm indication (RMAI) fault (RMAIFLT)

Register RMAIFLT counts RMAI faults that cause a PCM30 carrier to become system busy. RMAIFLT increases

- for each continuous RMAI error
- when RMAI errors that are not continuous reach the out-of-service limit (RMAIOL) and important data are in table LTCPSINV to make the carrier system busy
- for remote PM equipment failures

A continuous RMAI error is an error that persists for enough time to reach the out-of-service time limit (RMAIOST). The error count for errors that are not continuous is in the peripheral module (PM). Error count is reset every 5 min.

**Associated registers**

Register PCMCARR\_[RMAIERR](#) increases when remote equipment reports a multiframe level error, an equipment failure, or both in a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 either because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier becomes system busy.

**SIGLERR****Register type**

Peg

**Description**

Signaling channels error (SIGLERR)

Register SIGLERR increases when the system detects a transient change in the supervisory signaling channels of a PCM30 carrier.

**Associated registers**

Register [SIGFLT](#) counts transient change faults that the system detects in the supervisory signaling channels of a PCM30 carrier. The transient change faults make the carrier system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SIGFLT****Register type**

Peg

**Description**

Signaling channels fault (SIGFLT)

Register SIGLFLT counts transient change faults the system detects in the supervisory signaling channels. The transient change faults cause a PCM30 carrier to become system busy.

A PCM30 carrier becomes system busy if the transient changes detected in the supervisory signaling channels reach the out-of-service limit (SIGLOL). Table LTCPSINV must have data entered that allow the PCM30 carrier to become system busy.

**Associated registers**

Register [SIGLERR](#) increases when the system detects a transient change in the supervisory signaling channels of a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.

**SLIPERR****Register type**

Peg

**Description**

Slip error (SLIPERR)

Register SLIPERR increases when the system detects a frame slip in a PCM30 carrier.

**Associated registers**

Register [SLIPFLT](#) counts frame slip faults that make a PCM30 system busy.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**SLIPFLT****Register type**

Peg

**Description**

Slip fault (SLIPFLT)

Register SLIPFLT counts frame slip faults that cause a PCM30 carrier to become system busy.

A PCM30 carrier becomes system busy if its frame slips reach the out-of-service limit (SLIPOL in table CARRMTC). Table LTCPSINV must have data entered that allow PCM30 carrier to become system busy.

**Associated registers**

Register [SLIPERR](#) increases when the system detects a frame slip in a PCM30 carrier.

**Extension registers**

There are no extension registers.

**Associated logs**

The system generates PM110 when the system receives carrier alarms.

The system generates PM111 when a carrier returns to service from a system busy state.

The system generates PM180 because of improper software executions or because a hardware problem affects software execution.

The system generates PM186 when a carrier returns to service.

The system generates PM187 when a carrier is system busy.



## PCNF

### Description

Preset conference (PCNF)

The PCNF counts preset conferencing attempts on the integrated business network (IBN).

To initiate a conference with preset conferencing, the subscriber dials a preset number. The preset number causes the stations of preset conference members to ring at the same time. The preset dialing list holds a maximum of 25 conference members.

An attempt to dial a preset conference number that is not authorized triggers one of the following actions:

- The attendant intercepts if an IBN agent is the originator.
- The system routes the call to a tone or announcement.

The OM group PCNF provides one tuple for each conference. The following table lists the key and info fields associated with OM group PCNF.

| Key field                                  | Info field                                                                                        |
|--------------------------------------------|---------------------------------------------------------------------------------------------------|
| preset conference number in table PRECONF. | PCNF_CF6P_REQD is the number of six-port conference circuits required for each preset conference. |

### Related functional groups

There are no related functional groups.

### Registers

The following table lists the registers associated with OM group PCNF and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PCNF

| Register name           | Measures                  |
|-------------------------|---------------------------|
| <a href="#">PCNFATT</a> | Preset conference attempt |

**PCNFATT****Register type**

Peg

**Description**

Preset conference attempt (PCNFATT)

Register PCNFATT counts the attempts to activate the preset conference list.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## PJM

### Description

The PJM group measures the performance of the Persistence Job Manager (PJM). The PJM is a software subsystem responsible for performing database queries on behalf of call processing tasks. Each query is enqueued on the PJM's job queue as a Persistence Job.

The following table lists the key and info fields associated with OM group PJM.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

The following functional groups are related to OM group PJM:

- Session Manager

### Registers

The following table lists the registers associated with OM group PJM and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PJM

| Register name                 | Measures       |
|-------------------------------|----------------|
| <a href="#">queueHigh</a>     | queue high     |
| <a href="#">queueSize</a>     | queue size     |
| <a href="#">queueLow</a>      | queue low      |
| <a href="#">jobsAttempted</a> | jobs attempted |
| <a href="#">jobsServed</a>    | jobs served    |

#### queueHigh

##### Register type

Usage

##### Scan rate

**Description**

The high water mark for the queueSize register.

**Associated registers**

[queueSize](#), [queueLow](#)

**Extension registers**

None

**Associated logs**

OLC 402

**queueSize****Register type**

Watermark

**Scan rate****Description**

The number of persistence jobs currently in the PJM queue.

**Associated registers**

[queueHigh](#), [queueLow](#)

**Extension registers**

None

**Associated logs**

OLC 402

**queueLow****Register type**

Usage

**Scan rate****Description**

The low water mark for the queueSize register.

**Associated registers**

[queueHigh](#), [queueSize](#)

**Extension registers**

None

**Associated logs**

OLC 402

**jobsAttempted****Register type**

Peg

**Description**

The number of attempts to enqueue a job on the PJM queue.

**Associated registers**[jobsServed](#)**Extension registers**

None

**Associated logs**

None

**jobsServed****Register type**

Peg

**Description**

The number of successful attempts to enqueue a job on the PJM queue.

**Associated registers**[jobsAttempted](#)**Extension registers**

None

**Associated logs**

None

## Thresholding and Alarms

The queueSize register is associated with the OLC 402 alarm. This alarm is raised when the queueSize exceeds the positive-going threshold specified by the Overload.DBQueueThresholds engineering parameter and is cleared when the queueSize falls below the negative-going threshold specified by that same parameter.

## PKTMA

### Description

The Packet Media Anchor (PKTMA) OM group provides statistical information for PKTMA resources:

- anchored call attempts
- failed anchored call attempts
- maximum anchored call attempts (high water mark)

The following table lists the key and info fields associated with OM group PKTMA.

| Key field     | Info field                                                  |
|---------------|-------------------------------------------------------------|
| Integer       | PKTMA anchor number (derived from datafill order)           |
| PKTMA_OM_INFO | The maximum number of anchored calls supported in the CS2K. |

**Example:**

0 PKTMA 13

### Related functional groups

There are no functional groups associated with OM group PKTMA.

### Registers

The following table lists the registers associated with OM group PKTMA and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PKTMA

| Register name            | Measures                                                        |
|--------------------------|-----------------------------------------------------------------|
| <a href="#">PMAREQST</a> | number of anchored call attempts                                |
| <a href="#">PMAFLNR</a>  | number of failed call attempts due to unavailable resources     |
| <a href="#">PMAHWM</a>   | maximum number of simultaneous anchored calls (high water mark) |

**PMAREQST****Register type**

Peg

**Description**

Packet Media Anchor Requests (PMAREQST) counts the number of anchored call attempts.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PMAFLNR****Register type**

Peg

**Description**

Packet Media Anchor Failed No Resources (PMAFLNR) counts the number of failed anchored call attempts due to the unavailability of resources.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

Log XPKT340 is generated for failed anchored call attempts.



**PMAHWM****Register type**

Peg

**Description**

Packet Media Anchor High Water Mark (PMAHWM) indicates the maximum number of simultaneous anchored call attempts.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## PM

### Description

OM group Peripheral Module (PM) counts errors, faults, and maintenance state changes for DMS PMs with node numbers. PM performs separate counts for each PM that associates with a DMS switch. The data show the performance of PMs.

The following table lists the key and info fields associated with OM group PM.

| Key field             | Info field                                                                                                                                                                                   |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| There is no key field | PM_OM_INFO_TYPE includes PM node type, internal number of the node, and optional asterisk. Refer to <a href="#">Info field values and PM types for OM group PM</a> for the correct PM types. |

The following table lists the info field values and PM types for OM group PM.

#### Info field values and PM types for OM group PM (Sheet 1 of 7)

| Value | PM type                            |
|-------|------------------------------------|
| ADTC  | Austrian digital trunk controller  |
| ALCM  | Austrian line concentrating module |
| ALGC  | Austrian line group controller     |
| AP    | Application processor              |
| APU   | Application processing unit        |
| ARCC  | Austrian remote cluster controller |
| CFI   | Channel frame interface            |
| CFP   | Channel frame processor            |
| CSC   | Cell site controller               |
| STM   | Conference trunk module            |

**Info field values and PM types for OM group PM (Sheet 2 of 7)**

| <b>Value</b> | <b>PM type</b>                             |
|--------------|--------------------------------------------|
| DA           | Directory assistance database              |
| DCA          | Austrian digital carrier module            |
| DCM          | Digital carrier module                     |
| DCM250       | Digital carrier module DMS-250             |
| DES          | Digital echo suppressor                    |
| DFI          | Direct fiber interface                     |
| DLM          | Digital line module                        |
| DTC          | Digital trunk controller                   |
| DTC7         | CCS7 Digital trunk controller              |
| DTCI         | Digital trunk controller for ISDN          |
| DTCO         | Digital trunk controller offshore          |
| DTM          | Digital trunk module                       |
| EIU          | Ethernet interface unit                    |
| ELCM         | Enhanced line concentrating module         |
| ESA          | Emergency stand-alone                      |
| EXND         | External node                              |
| FRCC         | Force (download) remote cluster controller |
| FRIU         | Frame relay interface unit                 |
| FILP         | File processor                             |
| GIC          | Generic interface controller               |
| HFT          | HDLC frame transceiver                     |
| HSI          | High speed interface                       |
| HSI2         | High speed interface series 2              |

**Info field values and PM types for OM group PM (Sheet 3 of 7)**

| <b>Value</b> | <b>PM type</b>                                   |
|--------------|--------------------------------------------------|
| HSIE         | High speed interface extended                    |
| IAC          | ISDN access controller                           |
| ICP          | Integrated cellular peripheral                   |
| ICRM         | Integrated cellular remote module                |
| IDT          | Integrated digital terminal                      |
| IDTC         | International digital trunk controller           |
| ILCM         | International line concentrating module          |
| LGC          | International line group controller              |
| LTC          | International line trunk controller              |
| XLCM         | International extended line concentrating module |
| IPE          | Intelligent peripheral equipment                 |
| ITAC         | International TATS access controller             |
| LCM          | Line concentrating module                        |
| LCME         | Enhanced line concentrating module               |
| LCMI         | ISDN line concentrating module                   |
| LCOM         | LIU-COM (link interface unit data communication) |
| LDT          | Line appearance on a digital trunk               |
| LGC          | Line group controller                            |
| LGCI         | Line group controller ISDN                       |
| LGCO         | Line group controller offshore                   |
| LIM          | Link interface module                            |
| LIU          | Link interface unit                              |

**Info field values and PM types for OM group PM (Sheet 4 of 7)**

| <b>Value</b> | <b>PM type</b>                    |
|--------------|-----------------------------------|
| LIU7         | CCS7 link interface unit          |
| HLIU         | High-speed link interface unit    |
| HSLR         | High-speed link router            |
| LM           | Line module                       |
| LRU          | Line resource unit                |
| LTC          | Line trunk controller             |
| LTCI         | Line trunk controller ISDN        |
| MMA          | Austrian maintenance trunk module |
| MSB6         | Message switch buffer for CCIS6   |
| MSB7         | Message switch buffer for CCIS7   |
| MTM          | Maintenance trunk module          |
| NIU          | Network interface unit            |
| OAU          | Office alarm unit                 |
| OPM          | Outside plant module              |
| ORDB         | Operator reference database       |
| PDTC         | PCM30 digital trunk controller    |
| PLGC         | PCM30 line group controller       |
| PND          | PNODE                             |
| PRCC         | PCM30 remote cluster controller   |
| PSP          | Programmable signal processor     |
| PTM          | Packaged trunk module             |
| RCC          | Remote cluster controller         |
| RCC2         | Compact remote cluster controller |

**Info field values and PM types for OM group PM (Sheet 5 of 7)**

| <b>Value</b> | <b>PM type</b>                        |
|--------------|---------------------------------------|
| RCCI         | ISDN remote cluster controller        |
| RSCO2        | Remote switching center offshore 2    |
| RCS          | Remote concentrator SLC-96            |
| RCT          | Remote concentrator terminal          |
| RCU          | Remote carrier urban                  |
| RLC          | Remote line controller                |
| RLCM         | Remote line concentrating module      |
| RLM          | Remote line module                    |
| RMM          | Remote maintenance module             |
| RMSC         | Remote mobile switching center        |
| RSC          | Remote switching center               |
| RSCO         | Remote switching center offshore      |
| RSM          | Remote service module                 |
| SCM          | Subscriber carrier module             |
| SMA          | Subscriber module access              |
| SMR          | Subscriber carrier module-100 rural   |
| SMS          | Subscriber carrier module-100S        |
| SMSR         | Subscriber carrier module-100S remote |
| SMU          | Subscriber carrier module-100 urban   |
| SPM          | Service peripheral module             |
| SRCC         | SONET remote cluster controller       |
| SRU          | Small remote unit (ISDN LCM)          |
| STCM         | Signal terminal controller module     |

**Info field values and PM types for OM group PM (Sheet 6 of 7)**

| <b>Value</b> | <b>PM type</b>                            |
|--------------|-------------------------------------------|
| STM          | Service trunk module                      |
| STS          | Standardized traffic statistics           |
| SVR7         | CCS7 server                               |
| T8A          | Trunk module for CCITT circuits           |
| TACC         | TATS access controller                    |
| TAN          | Test access network                       |
| TDTC         | MOC DTC (MOC is an NT licensee)           |
| TLGC         | MOC LGC (MOC is an NT licensee)           |
| TLTC         | MOC LTC (MOC is an NT licensee)           |
| TM           | Trunk module                              |
| TM2          | Trunk module—two-wire                     |
| TM4          | Trunk module—four-wire                    |
| TM8          | Trunk module ATT testing                  |
| TMA          | Trunk module Austria                      |
| TMS          | TOPS message switch                       |
| TPC          | TOPS position controller                  |
| TRCC         | MOC RCC (MOC is a NT licensee)            |
| VLCM         | Virtual line concentrating module         |
| VPU          | Voice processing unit                     |
| VSR          | Very small remote                         |
| VSROM        | Very small remote                         |
| XLCM         | Expanded memory line concentrating module |

**Info field values and PM types for OM group PM (Sheet 7 of 7)**

| Value | PM type                                   |
|-------|-------------------------------------------|
| XLIU  | X.25/X.75 link interface unit             |
| XRLCM | Extended remote line concentrating module |

**Related functional groups**

The following functional groups are associated with OM group PM:

- DMS-100 local office
- DMS-100/200 combined local/toll office
- DMS-100/200 combined local/toll office with TOPS
- DMS-200 toll office
- DMS-200 with TOPS
- DMS-100 Meridian
- DMS-MTX mobile telephone exchange
- DMS-250 toll/tandem switch
- DMS-300 gateway
- Meridian 1 (options 111-211) PABX

**Registers**

The following table lists the registers associated with OM group PM and what they measure. For a description of a register, click on the register name.

**Registers for OM group PM (Sheet 1 of 2)**

| Register name           | Measures                         |
|-------------------------|----------------------------------|
| <a href="#">PMCCTDG</a> | PM circuit diagnostics run       |
| <a href="#">PMCCTFL</a> | PM circuit tests failed          |
| <a href="#">PMCCTOP</a> | PM circuit test outside plant    |
| <a href="#">PMDRERR</a> | PM drawer error                  |
| <a href="#">PMDRFLT</a> | PM drawer faults                 |
| <a href="#">PMDRMBU</a> | PM drawer manual busy (ManB) use |



**Registers for OM group PM (Sheet 2 of 2)**

| Register name                   | Measures                                     |
|---------------------------------|----------------------------------------------|
| <a href="#"><u>PMDRSBU</u></a>  | PM drawer system busy (SysB) usage           |
| <a href="#"><u>PMERR</u></a>    | PM error                                     |
| <a href="#"><u>PMFLT</u></a>    | PM fault                                     |
| <a href="#"><u>PMINTEG</u></a>  | PM accuracy failures                         |
| <a href="#"><u>PMMBP</u></a>    | PM changes to manual busy                    |
| <a href="#"><u>PMMBTCO</u></a>  | PM manual-busy terminals cut off             |
| <a href="#"><u>PMMCXFR</u></a>  | PM manual cold transfers                     |
| <a href="#"><u>PMMMBU</u></a>   | PM manual busy usage                         |
| <a href="#"><u>PMMSBU</u></a>   | PM system busy usage                         |
| <a href="#"><u>PMMWXFR</u></a>  | PM manual warm transfers                     |
| <a href="#"><u>PMPSEERR</u></a> | PM peripheral-side errors                    |
| <a href="#"><u>PMPSEFLT</u></a> | PM peripheral-side faults                    |
| <a href="#"><u>PMRGERR</u></a>  | PM ringing generator errors while in service |
| <a href="#"><u>PMRGFLT</u></a>  | PM ringing generator faults while in service |
| <a href="#"><u>PMSBP</u></a>    | PM changes to system busy                    |
| <a href="#"><u>PMSBTCO</u></a>  | PM system-busy terminals cut off             |
| <a href="#"><u>PMSCXFR</u></a>  | PM system cold transfers                     |
| <a href="#"><u>PMSWXFR</u></a>  | PM system warm transfers                     |
| <a href="#"><u>PMUMBU</u></a>   | Peripheral module unit manual-busy use       |
| <a href="#"><u>PMUSBU</u></a>   | Peripheral module unit system-busy use       |

**PMCCTDG****Register type**

Peg

**Description**

PMCCTDG counts system-initiated diagnostic tests of line or trunk cards. The tests examine problems that recur during call processing. Maintenance conditions that cause PMCCTDG to increase are different for each PM type.

For the digital carrier module (DCM), PMCCTDG counts the tests of a trunk interface card. The tests scan for removal of a DS-1 interface card and for a frame loss that causes a local or remote-carrier-group alarm state.

For the line module or the digital line module, PMCCTDG counts system-initiated tests of a line card.

For the trunk module (TM), PMCCTDG counts tests of any trunk interface card or service circuit. The TM tests:

- check that cards of the right type are present on the shelf
- test relay operation
- operation and release of signal distribution points and analysis of scan results
- transmission loss in looparound mode

For the extended multiprocessor system (XMS)-based peripheral modules (XPM), PMCCTDG increases when a system-initiated test runs on a line or trunk. The test runs because of repeated problems during call processing.

**Associated registers**

[PMCCTFL](#)

**Extension registers**

None

**Associated logs**

PM110, TRK106

**PMCCTFL****Register type**

Peg

**Description**

PMCCTFL increases when a system-initiated test finds a PM maintenance problem caused by a fault condition. Faults that increase the register differ for each PM type.

For the digital carrier module (DCM), PMCCTFL increases when tests show a fault caused by:

- the removal of a card
- a transmission error that results in a carrier group alarm

For the line module, PMCCTFL increases when tests reveal a maintenance problem caused by a PM, card or facility fault, or a wrong or missing card.

For the trunk module (TM), the digital carrier module (DCM), and XPM, PMCCTFL increases when tests detect a wrong card, or a missing or faulty card.

#### **Associated registers**

[PMCCTDG](#)

#### **Extension registers**

None

#### **Associated logs**

PM109, PM183, TRK106

### **PMCCTOP**

#### **Register type**

Peg

#### **Description**

PMCCTOP increases when system tests detect a fault on a line or trunk circuit located outside the switching office. Conditions that increase PMCCTOP vary with PM types. PMCCTOP increases the first time the fault appears, with all PM types. The register does not increase if the fault appears after new tests.

For the digital carrier module (DCM) and the trunk module (TM), PMCCTOP increases when the signaling-test system at a switching office detects a fault on a trunk circuit that is between the circuit and a far-end office. For example, PMCCTOP increases when an originating office does not receive a start-dial or wink signal. The far-end office sends these signals to the originating office in response to an off-hook signal.

For the line module, PMCCTOP increases when system tests detect a fault on a line circuit located outside the switching office.

For extended multiprocessor system (XMS)-based peripheral modules (XPM), PMCCTOP is incremented when system tests detect a fault on a line or trunk that is located outside the switching office.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PMDRERR****Register type**

Peg

**Description**

PMDRERR counts errors in a line drawer that cause the drawer to become in-service trouble.

**Associated registers**

[PMDRFLT](#)

**Extension registers**

None

**Associated logs**

PM102, PM181

**PMDRFLT****Register type**

Peg

**Description**

PMDRFLT counts faults in a line drawer that cause the drawer to become system busy.

**Associated registers**

[PMDRERR](#)

**Extension registers**

None

**Associated logs**

PM102, PM181

**PMDRMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMDRMBU is a usage register. Every 100 seconds, the system scans the line drawers in a PM and PMDRMBU records manual busy line drawers.

**Associated registers**[PMDRSBU](#)**Extension registers**

None

**Associated logs**

PM102, PM128

**PMDRSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMDRSBU is a use register. Every 100 seconds, the system scans line drawers in the PM, and PMDRSBU records system-busy line drawers.

**Associated registers**[PMDRMBU](#)**Extension registers**

None

**Associated logs**

PM102, PM128

**PMERR****Register type**

Peg

**Description**

PMERR counts errors in an in-service PM. The error conditions that cause PMERR to increase vary by PM type.

For Series-1 PMs, like line modules, digital carrier modules, and trunk modules, PMERR counts:

- command protocol violations
- RAM parity failures
- firmware errors
- controller message congestion
- test failures during a routine or initialization audit
- failures to respond to a message over either plane

For extended multiprocessor system (XMS)-based peripheral modules (for example, line concentrating modules, line group controllers, and line trunk controllers), PMERR counts:

- errors that only result in the generation of a log
- errors that result in additional maintenance action
- accuracy failures
- errors that result in who-am-I (WAI) messages
- changes from in-service to central-side (C-side) busy or system busy
- restart reports
- an event that causes a fault and increases register PMFLT

**Associated registers**

[PMFLT](#)

**Extension registers**

None

**Associated logs**

CCS231, CCS236, DDM101, DDM102, DDM104, DLC101, DPAC103, LOST108, LOST 109, LOST111, MPC906, NET102, NPAC210, PM101, PM108, PM113, PM115, PM116, PM117, PM118, PM119, PM121, PM122, PM124, PM125, PM126, PM128, PM150, PM160, PM180, PM181, PM194, PM198, TRK123, UTR100

**PMFLT****Register type**

Peg

**Description**

PMFLT counts faults that make a PM or one of its units system busy. PMFLT does not count the same fault again when repeated system tests attempt to clear the fault. Conditions that increase PMFLT differ between PMs and XPM.

For in-service trouble PMs, like line modules, trunk modules, and digital carrier modules, PMFLT counts errors that make the PM system busy. PMFLT counts these errors while the PM waits for manual or system recovery.

For XPMs like line concentrating modules, line group controllers, and line trunk controllers, PMFLT increases if:

- a complete PM or a single unit of a PM becomes system busy
- a central-side (C-side) node or link becomes manual busy, then returns to service. This condition results in a state change from C-side busy to system busy because the return to service tests failed during a system audit

**Associated registers**

[PMERR](#)

**Extension registers**

None

**Associated logs**

DLC102, DPAC104, MPC904, NPAC211, PM100, PM101, PM102, PM114, PM117, PM127, PM151, PM161, PM162, PM164, PM180, PM181, PM185, PM199

**PMINTEG****Register type**

Peg

**Description**

PMINTEG increases when the PM reports an accuracy failure to the central control.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

NET101, NET102, PM108, PM113, PM118, PM119, PM122, PM124, PM180, PM181, PM185, TRK122

**PMMBP****Register type**

Peg

**Description**

PMMBP increases when a PM becomes manual busy from an in-service or in-service trouble state. PMMBP increases when an LM becomes manual busy during manually requested warm and cold takeovers.

**Associated registers**

[PMSBP](#)

**Extension registers**

None

**Associated logs**

PM182, PM191

**PMMBTCO****Register type**

Peg

**Description**

PMMBTCO counts subscriber calls (terminals) that the system cuts off when a PM becomes manual busy. These calls associate with lines or trunks that are either call processing busy or call processing deloading.

**Associated registers**

[PMSBTCO](#)

**Extension registers**

None

**Associated logs**

None

**PMMCXFR****Register type**

Peg



**Description**

PMMCXFR increases when a manual action causes an XPM to perform a cold switch of activity (SWACT). Execution of the SWACT command at the MAP terminal can trigger a cold SWACT. A manual request that makes the active unit manual busy (while the inactive unit is in service) can also trigger a cold SWACT.

**Associated registers**

[PMSCXFR](#)

**Extension registers**

None

**Associated logs**

PM128, PM180

**PMMMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMMMBU is a usage register. Every 100 seconds, the system scans the PM and PMMMBU records manual-busy PMs.

**Associated registers**

[PMUMBU](#)

**Extension registers**

None

**Associated logs**

CCS218, CCS233, PM105, PM128, PM170, PM182, PM191

**PMMSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMMSBU is a usage register. Every 100 seconds, the system scans the PMs and PMMSBU records system-busy PMs. For dual-unit PMs,

PMMSBU increases one time if both units are system busy. PMMSBU also increases if one unit is system busy and the other unit is not in service. The hardware or software problems that make the PM system busy vary by PM type.

For a digital carrier module (DCM) or trunk module (TM), the following problems make the PM system busy:

- the DCM or TM fails a routine audit
- message paths are not available to the DCM or TM
- the DCM or TM sends more than 200 not requested trouble reports within one 10-minute audit period

For a line module (LM), any of the following problems make the PM system busy:

- the LM is not accessible
- the control section of the LM fails an audit
- the LM reports more than 200 controller errors or line errors between audits

#### **Associated registers**

[PMUSBU](#)

#### **Extension registers**

None

#### **Associated logs**

CCS219, CCS234, PM102, PM128, PM170, PM183, PM190, PM192

### **PMMWXFR**

#### **Register type**

Peg

#### **Description**

PMMWXFR increases if manual maintenance forces a dual-unit PM to perform a transfer of activity. A transfer that consists of either a warm SWACT or a unit takeover increases this register. The type of activity transfer depends on the type of PM that the manual request acts upon. PMMWXFR increases if:

- a manual request forces an XPM, like a line group controller or a line trunk controller, to perform a warm SWACT
- a manual request forces a line concentrating module (LCM) to perform a takeover of one unit by the other

To force an LCM to perform a takeover, make one unit of the LCM manual-busy while the mate unit is in service. PMMWXFR counts a takeover of one unit of an LCM by the other unit, but not a takeback of activity.

Examples of manual actions that can force an XPM to perform a warm SWACT are:

- the execution of the SWACT command at the MAP terminal
- a manual request that makes the active unit of an XPM manual busy while the inactive unit is in service

**Associated registers**

[PMSWXFR](#)

**Extension registers**

None

**Associated logs**

PM128, PM180

**PMPSEERR****Register type**

Peg

**Description**

PMPSEERR counts errors on the P-side interface of an XPM, or on a link interface module (LIM) frame transport bus (F-bus).

The register counts:

- errors in interface cards that terminate lines, trunks, or links
- errors in lines trunks or links
- F-bus errors

PMPSEERR increases if the error affects service or if it results in additional maintenance action. XPMs include the line concentrating module (LCM).

**Associated registers**

[PMPSEFLT](#)

**Extension registers**

None

**Associated logs**

PM110, PM181, PM183

**PMPSFLT****Register type**

Peg

**Description**

PMPSFLT counts faults on the P-side interface of an XPM or on the LIM frame transport bus (F-bus). The faults affect service and require more maintenance. The XPMs include the line concentrating module (LCM). PMPSFLT counts:

- faults in P-side interface cards that terminate trunks, lines, or links
- faults in lines, trunks, and links serviced by the interface cards
- faults in the F-bus

**Associated registers**[PMPSEERR](#)**Extension registers**

None

**Associated logs**

PM109, PM181, PM183

**PMRGERR****Register type**

Peg

**Description**

PMRGERR counts errors in the ringing generators that supply ringing and automatic number identification (ANI) coin functions to the line concentrating module (LCM). PMRGERR counts all ringing generator errors, even if the ringing generator is not in service at the time of the error. The LCM must be in service at the time of the error.

A single ringing generator can service both LCMs in the same frame. The register can count one ringing generator error four times. The count notes each of the two line concentrating arrays in each of the two LCMs.

**Associated registers**[PMRGFLT](#)

**Extension registers**

None

**Associated logs**

PM160

**PMRGFLT****Register type**

Peg

**Description**

PMRGFLT counts service-affecting faults detected in the ringing generators that supply ringing and ANI coin functions to the LCM. The ringing generator must be in service for PMRGFLT to increase.

On Meridian SL-100 switches, the Intelligent Peripheral Equipment (IPE) counts analog phone ring failures that are due to an overloaded ring generator.

**Associated registers**

[PMRGERR](#)

**Extension registers**

None

**Associated logs**

PM161, PM162, PM163, PM189

**PMSBP****Register type**

Peg

**Description**

PMSBP increases when the an in-service or in-service trouble PM becomes system busy. A PM usually becomes central-side (C-side) busy before system busy. If the PM returns to service from the C-side busy state and does not become system busy, PMSBP does not increase.

For line modules (LM), PMSBP increases when the LM becomes system busy during both warm and cold takeovers.

**Associated registers**

[PMSBP](#)

**Extension registers**

None

**Associated logs**

PM107, PM183, PM190, PM192

**PMSBTCO****Register type**

Peg

**Description**

PMSBTCO counts subscriber calls (terminals) cut off when the PM becomes system busy. Conditions that increase PMSBTCO vary with different PMs. The register counts subscriber calls for lines or trunks that are call-processing busy or call-processing deloading.

PMSBTCO counts subscriber calls cut off when the PM state changes to central-side (C-side) busy from in-service or in-service trouble. PMSBTCO counts the calls for the digital carrier module and the trunk module. C-side busy is an intermediate state that occurs before the PM becomes system busy.

PMSBTCO counts the subscriber calls cut off when a LM becomes system busy. An LM can recover from the C-side busy state and become system busy when the mate LM becomes system busy.

PMSBTCO for the recovered LM increases by the number of subscriber calls that the system busy mate cuts off. The increase occurs when an LM performs a cold takeover. The LM becomes responsible for the calls of the mate LM but cannot preserve these calls through the takeover.

If a warm takeover occurs when an LM becomes system busy, calls are not cut off and PMSBTCO does not increase. PMSBTCO increases when an LM returns to service from system busy. As the LM returns to service, the LM performs a warm takeback of control of its line drawers. The increase is equal to the number of calls that the original change to system busy cutoff.

For XPMs, PMSBTCO counts the subscriber calls cut off when the PM becomes system busy. PMSBTCO increases when a call in the talking state is cut off.

**Associated registers**

[PMMBTCO](#)

**Extension registers**

None

**Associated logs**

None

**PMSCXFR****Register type**

Peg

**Description**

PMSCXFR increases when a system action causes an XPM to perform a cold SWACT. Examples of system actions that trigger a cold SWACT:

- an XPM forced to perform a cold SWACT
- when the active unit of an XPM becomes system busy
- when the central-side (C-side) links to the active unit of an XPM becomes system busy

**Associated registers**

[PMMCXFR](#)

**Extension registers**

None

**Associated logs**

PM128, PM179, PM180, PM181

**PMSWXFR****Register type**

Peg

**Description**

PMSWXFR increases if system maintenance forces a dual-unit PM to perform a transfer of activity (warm SWACT or a unit takeover). The activity transfer depends on the type of PM that the system request acts on. PMSWXFR increases when:

- the system forces an XPM, such as a line group controller or line trunk controller, to perform a warm SWACT
- the system forces an LCM to perform a takeover of one unit by the other

PMSWXFR counts a takeover of one unit of the LCM by the other unit. The register does not count a takeback of activity in the LCM.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PM128, PM179, PM180, PM181

**PMUMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMUMBU is a use register. Every 100 seconds, the system scans the PMs and PMUMBU records the number of times a PM unit is manual busy. The register increases when a PM unit is set to manual busy and in each of the next scan intervals when the unit remains manual busy.

**Associated registers**

[PMUSBU](#), [PMMMBU](#), [PMMSBU](#)

**Extension registers**

None

**Associated logs**

PM105, PM128

**PMUSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PSUMBU is a use register. Every 100 seconds, the system scans the PMs and PMUSBU records the number of times a PM unit is system busy. The register increases when a PM unit is set to system busy, and in each of the next scan intervals when the unit remains system busy.

**Associated registers**

[PMUMBU](#), [PMMMBU](#), [PMMSBU](#)

**Extension registers**

None



**Associated logs**  
PM102, PM128

## PM1

### Description

Peripheral module single-unit maintenance summary (PM1)

The OM group PM1 provides information on the following: errors, faults, and system- and manual-busy use for single-unit peripheral modules (PM) without node numbers.

The OM group PM1 supplies the data that shows the performance of PM groups.

The OM group PM1 provides one tuple for each PM type defined in the key field. The following table lists the key and info fields associated with OM group PM1.

| Key field                                                                                                                                                                        | Info field                                                                                         |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| PM1_OMTYPE. This field consists of any of the following values: ST6OM, ST7OM, DCHOM, PH1OM, LIU7OM, DCHBX02OM, FRIUOM, EIUOM, APUOM, LCOMOM, XLIUOM, VPUOM, CAUOM, CIUOM, CAVUOM | PM1_OMINFO. This field contains the number of peripherals of the type identified in the key field. |

### Related functional groups

The following are the associated functional groups for OM group PM1:

- Automated Directory Assistance Service (ADAS)
- DMS-100 local office
- DMS-100/200 combined local/toll office
- DMS-100/200 combined local/toll office with TOPS
- DMS-200 toll office
- DMS-200 with TOPS
- DMS-100 Meridian
- DMS-MTX mobile telephone exchange
- DMS-250 toll/tandem switch
- DMS-300 gateway
- Meridian 1 (options 111-211) PABX

## Registers

The following table lists the registers associated with OM group PM1 and what they measure. For a description of a register, click on the register name.

### Registers for OM group PM1

| Register name            | Measures                                |
|--------------------------|-----------------------------------------|
| <a href="#">PM1ERR</a>   | PM single-unit errors                   |
| <a href="#">PM1FLT</a>   | PM single-unit fault                    |
| <a href="#">PM1INITS</a> | PM single-unit initializations          |
| <a href="#">PM1LOAD</a>  | PM single-unit reload required          |
| <a href="#">PM1MBU</a>   | PM single-unit manual-busy usage        |
| <a href="#">PM1PSERR</a> | PM single-unit P-side errors            |
| <a href="#">PM1PSFLT</a> | PM single-unit P-side faults            |
| <a href="#">PM1PSMBU</a> | PM single-unit P-side manual-busy usage |
| <a href="#">PM1PSSBU</a> | PM single-unit P-side system-busy usage |
| <a href="#">PM1SBU</a>   | PM single-unit system-busy usage        |

### PM1ERR

#### Register type

Peg

#### Description

PM single-unit errors (PM1ERR)

Register PM1ERR counts system-detected errors that an in-service PM reports. The PM1 increases when one of the following events occurs:

- a PM sends an unsolicited message that indicates a correct fault condition
- system-requested diagnostics remove the PM from service

#### Associated registers

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM190 appears in two formats. The first format appears when a system request makes a signaling terminal controller (STC) system busy. As a result, the signaling terminal (ST) that the log report identifies becomes system busy. The second format appears when a fault in the D-channel handler (DCH) makes the DCH system busy. The ISDN service group (ISG) defines the services that switch to a spare DCH, if a DCH is available, to prevent loss of service.

The system generates PM198 when an STC or a D-channel handler (DCH) sends an unsolicited message that indicates a legitimate fault condition. The fault condition does not affect service. System action should resolve the condition.

The system generates PM199 when an STC or a DCH ends a system-initiated diagnostic test. The result of the diagnostic test appears in the log.

**PM1FLT****Register type**

Peg

**Description**

PM single-unit fault (PM1FLT)

Register PM1FLT increases when the system removes a PM from service because of a continuing fault that system-initiated diagnostics detect.

Register PM1FLT counts the faults for all PM cards except P-side and C-side interface cards. The register counts each fault one time.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM190 appears in two formats. The first format appears when a system request makes a signaling terminal controller (STC) system busy. As a result, the signaling terminal (ST) that the log report identifies becomes system busy. The second format appears when a fault in the D-channel handler (DCH) makes the DCH system busy. The

ISG defines the services that switch to a spare DCH, if a DCH is available, to prevent loss of service.

Log PM192 appears in two formats. The first format appears when the STC becomes manual busy, and the C-side node (the ISDN access controller [IAC]) goes out of service. The second format appears when the IAC of the DCH goes out of service.

The system generates PM194 when an STC or DCH:

- detects conditions that are not normal. The conditions are not hardware-related or are not linked to a hardware fault
- changes from an in-service state to an in-service trouble state

Log PM198 appears when an ST or a DCH sends an unsolicited message that indicates a correct fault condition. The fault condition does not affect service. System action should resolve the condition.

PM199 appears when either an STC or a DCH ends a system-initiated diagnostic test. The result of the diagnostic test appears in the log.

## **PM1INITS**

### **Register type**

Peg

### **Description**

PM single-unit initializations (PM1INITS)

The system does not support PM1INITS. The value is always zero.

### **Associated registers**

There are no associated registers.

### **Extension registers**

There are no extension registers.

### **Associated logs**

There are no associated logs.

## **PM1LOAD**

### **Register type**

Peg

### **Description**

PM single-unit reload required (PM1LOAD)

Register PM1LOAD is not supported. The value is always zero.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM1MBU****Register type**

Usage

**Description**

PM single-unit manual-busy usage (PM1MBU)

Register PM1MBU is a usage register. Every 10 s, the system scans the PMs and PM1MBU records manual-busy PMs.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

Log ISDN104 appears when the Bd channel goes out of service because of a loss of synchronization. The loss of synchronization occurs when:

- the D-channel handler goes out of service
- the DS-1 link goes out of service
- a problem occurs with the packet handler

Log PM191 appears in two formats. The first format appears when a manual request changes an STC to manual busy. As a result, the ST identified in PM191 becomes manual busy. The second format appears when a manual request changes the DCH to manual busy. The ISG field in PM191 identifies the services that this action affects.

**PM1PSERR****Register type**

Peg

**Description**

PM single-unit P-side errors (PM1PSERR)

Register PM1PSERR is inactive.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM1PSFLT****Register type**

Peg

**Description**

PM single-unit P-side faults (PM1PSFLT)

Register PM1PSFLT is inactive.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM1PSMBU****Register type**

Peg

**Description**

PM single-unit P-side manual-busy usage (PM1PSMBU)

Register PM1PSMBU is not active.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM1PSSBU****Register type**

Peg

**Description**

PM single-unit P-side system-busy usage

Register PM1PSSBU is not active.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM1SBU****Register type**

Peg

**Description**

PM single-unit system-busy usage (PM1SBU)

Register PM1SBU is a usage register. Every 10 s, the system scans the PMs, and PM1SBU records system-busy PMs.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

Log ISDN104 appears when the Bd channel goes out of service because of a loss of synchronization. The loss of synchronization occurs when

- the D-channel handler goes out of service
- the DS-1 link goes out of service
- a problem occurs with the packet handler

Log PM190 appears in two formats. The first format appears when a system requests changes an STC to system busy. As a result, the ST identified in the log report becomes system busy. The second format



appears when a fault in the DCH makes the DCH system busy. The ISG defines the services that switch to a spare DCH, if a DCH is available, to prevent loss of service.

---

## PM2

---

### Description

Dual-unit peripheral module maintenance summary (PM2)

The OM group PM2 provides information on the performance of dual-unit peripheral modules (PM) of type IPML (without node numbers). The PM2 also collects data for the single-unit very small remote (VSR) PMs.

The PM2 has 20 registers that count the following events:

- PM errors and faults
- unit initializations
- unit reloads
- control transfers
- emergency control transfers
- system- or manual-busy lines
- line errors and faults
- manual and system warm and cold control transfers
- terminals cut off by manual- or system-busy PMs
- peripheral side (P-side) errors and faults
- ringing generator errors and faults

PM2 has four usage registers that record when

- a PM unit is manual or system busy
- a PM is manual or system busy

The data from PM2 show the performance of dual-unit PMs and the single-unit VSR.

The OM group PM2 provides one tuple for each key. The following table lists the key and info fields associated with OM group PM2.

| Key field                                                                                                                                                                                                                                                                                         | Info field                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| PM2_OMTYPE is a field that consists of any of the following values: ADTC, ALGC, ARCC, DFI, DLM, DTC, DTCI, ELCM, ESA, GIC, HSI2, IAC, ICP, IDTC, ILCM, ILGC, ILTC, IPML, LCM, LGC, LTC, MSB6, MSB7, PDTC, PLGC, PRCC, RC02, RCC, RCC2, RCS, RCT, SMS, SMSR, SMU, SRCC, SRM, TDTC, TLGC, TMS, TRCC | PM2_OMINFO is a value that indicates the number of PMs of the type defined by the key. |

## Related functional groups

The following are associated functional groups for OM group PM2:

- DMS-100 local
- DMS-100/200
- DMS-100/200 TOPS
- DMS-200 toll
- DMS-200 TOPS
- DMS-MTX
- DMS-250
- DMS-300
- Meridian 1 (options 111-211) PBX

## Registers

The following table lists the registers associated with OM group PM2 and what they measure. For a description of a register, click on the register name.

### Registers for OM group PM2

| Register name            | Measures                                  |
|--------------------------|-------------------------------------------|
| <a href="#">PM2CCTER</a> | Peripheral module dual-unit circuit error |
| <a href="#">PM2CCTFL</a> | Peripheral module dual-unit circuit fault |

**Registers for OM group PM2**

| <b>Register name</b>     | <b>Measures</b>                                           |
|--------------------------|-----------------------------------------------------------|
| <a href="#">PM2CCTMB</a> | Peripheral module dual-unit circuit manual busy           |
| <a href="#">PM2CCTSB</a> | Peripheral module dual-unit circuit system busy           |
| <a href="#">PM2CXFR</a>  | Peripheral module dual-unit control transfer              |
| <a href="#">PM2ECXFR</a> | Peripheral module dual-unit emergency control transfer    |
| <a href="#">PM2ERR</a>   | Peripheral module dual-unit errors                        |
| <a href="#">PM2FLT</a>   | Peripheral module dual-unit faults                        |
| <a href="#">PM2INITS</a> | Peripheral module dual-unit initializations               |
| <a href="#">PM2LOAD</a>  | Peripheral module dual-unit load                          |
| <a href="#">PM2MBTCO</a> | Peripheral module dual-unit manual-busy terminals cut off |
| <a href="#">PM2MCXFR</a> | Peripheral module dual-unit manual cold transfers         |
| <a href="#">PM2MMBU</a>  | Peripheral module dual-unit module manual-busy use        |
| <a href="#">PM2MSBU</a>  | Peripheral module dual-unit module system-busy usage      |
| <a href="#">PM2MWXFR</a> | Peripheral module dual-unit manual warm transfers         |
| <a href="#">PM2PSERR</a> | Peripheral module dual-unit peripheral-side errors        |
| <a href="#">PM2PSFLT</a> | Peripheral module dual-unit peripheral-side faults        |
| <a href="#">PM2RGERR</a> | Peripheral module dual-unit ringing generator errors      |
| <a href="#">PM2RGFLT</a> | Peripheral module dual-unit ringing generator fault       |
| <a href="#">PM2SBTCO</a> | Peripheral module dual-unit system-busy terminals cut off |

**Registers for OM group PM2**

| Register name            | Measures                                          |
|--------------------------|---------------------------------------------------|
| <a href="#">PM2SCXFR</a> | Peripheral module dual-unit system cold transfers |
| <a href="#">PM2SWXFR</a> | Peripheral module dual-unit system warm transfer  |
| <a href="#">PM2UMBU</a>  | Peripheral module dual-unit manual-busy usage     |
| <a href="#">PM2USBU</a>  | Peripheral module dual-unit system-busy usage     |

**PM2CCTER****Register type**

Peg

**Description**

Peripheral module dual-unit circuit error (PM2CCTER)

Register PM2CCTER increases when an error appears in a PM terminal (line card).

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2CCTFL****Register type**

Peg

**Description**

Peripheral module dual-unit circuit fault (PM2CCTFL)

Register PM2CCTFL increases when a fault appears in a PM terminal (line card).

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2CCTMB****Register type**

Peg

**Description**

Peripheral module dual-unit circuit manual busy (PM2CCTMB)

PM2CCTMB increases when a PM terminal (line card) becomes manual busy.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2CCTSB****Register type**

Peg

**Description**

Peripheral module dual-unit circuit system busy (PM2CCTSB)

Register PM2CCTSB increases when a PM terminal (line card) becomes system busy.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2CXFR****Register type**

Peg

**Description**

Peripheral module dual-unit control transfer (PM2CXFR)

Register PM2CXFR increases when a PM unit changes activity from active to inactive for any reason. The mate unit takes control of the lines for the complete PM.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2ECXFR****Register type**

Peg

**Description**

Peripheral module dual-unit emergency control transfer (PM2ECXFR)

Register PM2ECXFR increases when a PM unit changes activity from active to inactive because of a fault or a manual uncontrolled transfer. The mate unit takes control of the lines for the complete PM.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2ERR****Register type**

Peg

**Description**

Peripheral module dual-unit errors (PM2ERR)

Register PM2ERR increases when an error occurs in either unit of an in-service PM. The error can cause additional maintenance action for the register to increase. Register PM2ERR counts the following events

- errors that only result in the generation of a log
- errors that result in additional maintenance action
- accuracy failures for PMs (except the line concentrating module)
- errors resulting in who-am-I (WAI) messages
- state changes from in-service to central-side (C-side) busy or in-service to system busy
- restart reports from the central control
- any event that causes a fault and increments PM2FLT

**Associated registers**

Register [PM2FLT](#) increases when a fault in either unit of an in-service PM makes the unit or the PM system busy.

Register PMTYP\_PMTERR counts the total errors that the system detects in a group of PMs of the same type. This count applies to very small remotes, and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM179 appears after detection of loss of talk battery on an LCM shelf.

Log PM180 appears when a software exception occurs.

**PM2FLT****Register type**

Peg

**Description**

Peripheral module dual-unit faults (PM2FLT)



Register PM2FLT increases when a fault in either unit of an in-service PM makes the unit or the PM system busy. Register PM2FLT increases when:

- a PM or PM unit becomes system busy
- a central-side (C-side) node or link becomes manual busy and returns to service. The result is a change from C-side busy to system busy

**Associated registers**

There are no associated registers.

Register [PM2ERR](#) increases when an error occurs in either unit of an in-service PM.

Register PMTYP\_PMFLT counts the PM faults detected in a group of PMs of the same type. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM181 appears when a PM exception occurs.

**PM2INITS****Register type**

Peg

**Description**

Peripheral module dual-unit initializations (PM2INITS)

Register PM2INITS increases when an in-service PM unit resets without a warning.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2LOAD****Register type**

Peg

**Description**

Peripheral module dual-unit load (PM2LOAD)

Register PM2LOAD increases when an in-service PM unit requires a reload that the central control did not request.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2MBTCO****Register type**

Peg

**Description**

Peripheral module dual-unit manual-busy terminals cut off (PM2MBTCO)

Register PM2MBTCO counts the subscriber calls (terminals) cut off when a PM becomes manual busy. The register only counts subscriber calls that associate with a line or trunk that is call-processing-busy or call-processing-deloding.

Register PM2MBTCO increases one time for two-port calls.

**Associated registers**

Register [PM2SBTCO](#) counts the subscriber calls (terminals) cut off when a PM becomes system busy.

Register PMTYP\_PMTMBTCO counts the subscriber calls cut off when a PM becomes manual busy. Register PMTYP\_PMTMBTCO collects data for a group of PMs of the same type. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2MCXFR****Register type**

Peg

**Description**

Peripheral module dual-unit manual cold transfers (PM2MCXFR)

Register PM2MCXFR increases when a manual request causes an extended multiprocessor system (XMS)-based peripheral module (XPM) to perform a cold switch of activity (SWACT). This action results from either of the following events:

- execution of the SWACT command at the MAP terminal
- the active unit becomes manual busy when the inactive unit is in service

**Associated registers**

Register [PM2SCXFR](#) increases when a system request causes an XPM to perform a cold SWACT.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2MMBU****Register type**

Usage

**Description**

Peripheral module dual-unit module manual-busy use (PM2MMBU)

Register PM2MMBU is a usage register. Every 10 s, the system scans the PMs and PM2MMBU records manual-busy PMs.

Register PM2UMBU can detect a PM that is manually- busy, because of the sampling rate.

**Associated registers**

Register [PM2MSBU](#) records system-busy PMs.

Register PMTYP\_PMTMSBU records when a group of PMs of the same type is manually-busy. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2MSBU****Register type**

Usage

**Description**

Peripheral module dual-unit module system-busy usage (PM2MSBU)

Register PM2MSBU is a usage register. Every 10 s, the system scans the PMs. The PM2MSBU records when both units of the PM are system busy. The register also records when one unit is system busy while the mate unit is out of service.

Register PM2UMBU can detect a PM that is system busy, because of the sampling rate.

**Associated registers**

Register [PM2MMBU](#) records manual-busy PMs.

Register PMTYP\_PM2MSBU records when a group of PMs of the same type is system busy. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM102 appears when a PM becomes system busy.

Log PM128 appears when a PM changes to in-service trouble because of system or manual action.

**PM2MWXFR****Register type**

Peg

**Description**

Peripheral module dual-unit manual warm transfers (PM2MWXFR)

PM2MWXFR increases when manual interruption causes either:

- an XPM to perform a warm SWACT, or
- an LCM to perform a takeover

The following manual activities cause an XPM to perform a warm SWACT:

- execution of the SWACT command
- the active unit becomes manual busy while the mate is in service

If one unit of an LCM becomes manual busy while the mate is in service, a takeover occurs. Register PM2MWXFR does not increase if an LCM takeback of activity occurs.

#### **Associated registers**

Register [PM2SWXFR](#) increases when system maintenance causes a warm transfer.

Register PMTYP\_PMTMWXFR increases when manual maintenance causes a dual-unit PM in a group of PMs of the same type to perform a SWACT. This count applies to enhanced line concentrating modules.

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

There are no associated logs.

### **PM2PSERR**

#### **Register type**

Peg

#### **Description**

Peripheral module dual-unit peripheral-side errors (PM2PSERR)

Register PM2PSERR counts errors detected on the P-side interface of a PM

Register PM2PSERR increases when one of the following errors occurs:

- errors that originate in interface cards that terminate P-side lines, trunks, or links, or
- P-side line, trunk, or link errors

**Associated registers**

Register [PM2PSFLT](#) counts errors detected on the P-side interface of a PM.

Register PMTYP\_PMTPSERR increases when an error appears on the P-side interface of a PM. Register PMTYP\_PMTPSERR collects data for a group of PMs of the same type. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2PSFLT****Register type**

Peg

**Description**

Peripheral module dual-unit peripheral-side faults (PM2PSFLT)

Register PM2PSFLT counts errors detected on the P-side interface of a PM.

Register PM2PSFLT increases when one of the following faults occur:

- faults that originate in P-side interface cards that terminate lines, trunks, or links
- P-side line, trunk, or link faults

**Associated registers**

Register [PM2PSERR](#) counts errors that the system detects on the P-side interface of a PM.

Register PMTYP\_PMTPSFLT increases when a fault appears on the P-side interface of a PM. The PMTYP\_PMTPSFLT collects data for a group of PMs of the same type. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## PM2RGERR

### Register type

Peg

### Description

Peripheral module dual-unit ringing generator errors (PM2RGERR)

Register PM2RGERR counts errors that the system detects in ringing generators. Ringing generators supply ringing and automatic number identification (ANI) coin functions to an in-service PM. The state of the ringing generator is not important.

A single ringing generator can service two line concentrating modules (LCM) in the same frame. Register PM2RGERR counts one ringing generator error four times if the operator services two modules. The register counts one time for each of the two line concentrating arrays in each LCM.

### Associated registers

Register [PM2RGFLT](#) counts faults that the system detects in ringing generators that supply ringing and ANI coin functions to a PM.

Register PMTYP\_PMTRGERR counts errors in the ringing generators that supply ringing and ANI coin functions to a PM. Register PMTYP\_PMTRGERR collects data for a group of PMs of the same type. This count applies to very small remotes.

### Extension registers

There are no extension registers.

### Associated logs

There are no associated logs.

## PM2RGFLT

### Register type

Peg

### Description

Peripheral module dual-unit ringing generator fault (PM2RGFLT)

Register PM2RGFLT counts faults in ringing generators that supply ringing and ANI coin functions to a PM. Register PM2RGFLT only counts faults in-service ringing generators.

**Associated registers**

Register [PM2RGERR](#) counts errors in ringing generators that supply ringing and ANI coin functions to an in-service PM.

Register PMTYP\_PMTRGFLT counts service-affecting faults in the ringing generators that supply ringing and ANI coin functions to a PM. The PMTYP\_PMTRGFLT collects data for a group of PMs of the same type. This count applies to very small remotes.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2SBTCO****Register type**

Peg

**Description**

Peripheral module dual-unit system-busy terminals cut off (PM2SBTCO)

Register PM2SBTCO counts subscriber calls (terminals) cut off when a PM becomes system busy. The register counts subscriber calls that associate with a line or trunk that is call processing busy or call processing deloading.

Register PM2MBTCO increases one time for two-port calls.

**Associated registers**

Register [PM2MBTCO](#) counts the subscriber calls (terminals) cut off when a PM becomes manual busy.

Register PMTYP\_PMTSBTCO counts the subscriber calls (terminals) cut off when a PM becomes system busy. Register PMTYP\_PMTSBTCO collects data for a group of PMs of the same type. This count applies to very small remotes and enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



**PM2SCXFR****Register type**

Peg

**Description**

Peripheral module dual-unit system cold transfers (PM2SCXFR)

Register PM2SCXFR increases when a system request causes an XPM to perform a cold SWACT through any of the following activities:

- the system requests a SWACT
- the active unit becomes system busy when the inactive unit is in service
- central-side (C-side) links to the active unit close while the inactive unit is in service

**Associated registers**

Register [PM2MCXFR](#) increases when a manual request causes an XPM to perform a cold SWACT.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2SWXFR****Register type**

Peg

**Description**

Peripheral module dual-unit system warm transfer (PM2SWXFR)

Register PM2SWXFR increases when system maintenance causes an XPM to perform a warm SWACT or an LCM to perform a takeover.

The system causes a warm SWACT in an XPM by one of the following actions:

- the system requests a warm SWACT
- the active unit becomes system busy
- central side (C-side) links to the active unit close while the inactive unit is in service

The system causes a takeover of a unit in an LCM in two ways. One unit becomes system busy while the mate is in service. A unit can become busy while C-side links to one unit close while the mate is in service.

Register PM2SWXFR does not increase if an LCM takeback of activity occurs.

**Associated registers**

Register [PM2MWXFR](#) increases when manual maintenance causes a warm transfer.

Register PMTYP\_PMTSWXFR increases when a system action causes a PM in a group of PMs of the same type to perform a SWACT. This count applies to enhanced line concentrating modules.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PM2UMBU****Register type**

Usage

**Description**

Peripheral module dual-unit manual-busy usage (PM2UMBU)

Register PM2UMBU is a usage register. Every 10 s, the system scans the PMs and PM2UMBU records manual-busy PM units.

Register PM2UMBU increases twice if both units of a PM are manual busy. Register PM2UMBU can detect a unit that is manual busy, because of the sampling rate.

**Associated registers**

Register [PM2USB](#) records system-busy PM units.

Register PMTYP\_PMTMMBU records when a group of PMs of the same type are manual busy. This count applies to the enhanced line concentrating module.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM105 appears when a PM becomes manual busy.

Log PM128 appears when a PM changes to in-service trouble because of system or manual action.

## **PM2USB**

### **Register type**

Usage

### **Description**

Peripheral module dual-unit system-busy usage (PM2USB)

Register PM2USB is a usage register. Every 10 s, the system scans the PMs and PM2USB records system-busy PM units.

Problems that make a PM unit system busy include:

- test failure
- too many unsolicited messages
- auto unit resets

Register PM2USB increases twice if both units of a PM are system busy. Register PM2USB can detect a unit that is system busy because of the sampling rate.

### **Associated registers**

Register [PM2UMB](#) records when a PM unit is manual busy.

Register PTMTY\_PMTUSB records when the units of a group of PMs of the same type are system busy. This count applies to the enhanced line concentrating modules.

### **Extension registers**

There are no extension registers.

### **Associated logs**

Log PM102 appears when a PM becomes system busy.

Log PM128 appears when a PM changes to in-service trouble because of system or manual action.

---

## PMMSGCNT

---

### Description

Peripheral module message counter (PMMSGCNT)

The PMMSGCNT provides information from the maintenance counters in the line concentrating module (LCM).

The maintenance counters count the following:

- messages from the LCM or the remote LCM (RLCM) to the host XMS-based peripheral module (XPM)
- messages from the host XPM to the LCM or RLCM
- information on the performance of the DMSX protocol

To show the contents of the counters at the peripheral module (PM) level of the MAP display, post an LCM and issue the QUERYPM command.

The user polls the LCMs and collects the information contained in the LCM maintenance counters. The user can also output the information in the OM group PMMSGCNT.

The PMMSGCNT contains 19 registers that count the following:

- wait-for-send timeouts on messages from the LCM to the C-side XPM
- wait-for-acknowledgement timeouts on messages from the LCM to the C-side XPM
- wait for link idle messages that the system receives after a negative acknowledgement on message transfer
- single negative acknowledgements that the system receives from the C-side XPM
- double negative acknowledgements that the system receives from the C-side XPM
- wait-for-start-of-message timeouts on messages from the C-side XPM to the LCM
- wait-for-idle messages from the C-side XPM to the LCM after a message transfer
- messages the LCM receives from the C-side XPM that have wrong cyclic redundancy check (CRC)

- messages from the C-side XPM to the LCM with more bytes than the system permits
- null messages the system receives from the C-side XPM that are not reset messages
- spurious frame interrupts
- messages the LCM receives from the C-side XPM that have an invalid node number
- messages that the LCM receives from the C-side XPM
- messages that the LCM transmits to the C-side XPM
- negative acknowledgements that the inter-unit communication (IUC) link receives
- negative acknowledgements that the IUC link receives because of invalid characters
- negative acknowledgements that the IUC link receives because of invalid byte counts
- negative acknowledgements that the IUC link receives because of wrong checksums
- negative acknowledgements that the IUC link receives because of invalid messages

The OM group PMMSGCNT provides one tuple for each line concentrating module unit. The following table lists the key and info fields associated with OM group PMMSGCNT.

| Key field | Info field                           |
|-----------|--------------------------------------|
| none      | PM_MSG_OMINFO is the LCM identifier. |

The LCM identifier consists of the site name, the frame number, the bay number, and the unit number.

The site name consists of four alphanumeric characters. An example is HOST.

- The frame number is zero to 99.
- The bay number is zero to 99.
- The unit number is 0 or 1.

Office parameter LCM\_PM\_MSG\_CNT in table OFCOPT is set to Y (yes). This office parameter is set to Y to activate the process of polling

the LCMs for the maintenance counters. This office parameter is also set to Y to output OM group PMMSGCNT.

## Related functional groups

The functional group LCM associates with OM group PMMSGCNT.

## Registers

The following table lists the registers associated with OM group PMMSGCNT and what they measure. For a description of a register, click on the register name.

### Registers for OM group PMMSGCNT

| Register name            | Measures                                                |
|--------------------------|---------------------------------------------------------|
| <a href="#">DNACK</a>    | Double negative acknowledgements                        |
| <a href="#">IDLSTATE</a> | Spurious frame interrupts                               |
| <a href="#">IINVBYTE</a> | Invalid byte                                            |
| <a href="#">IINVCHAR</a> | Invalid character                                       |
| <a href="#">IINVCKSM</a> | Invalid checksum                                        |
| <a href="#">IINVMSG</a>  | Invalid message                                         |
| <a href="#">INACK</a>    | Inter unit communication link negative acknowledgements |
| <a href="#">INVNODE</a>  | Invalid node                                            |
| <a href="#">NULLMSG</a>  | Null messages                                           |
| <a href="#">PMCRC</a>    | Not correct cyclic redundancy check                     |
| <a href="#">PMNACK</a>   | Single negative acknowledgement                         |
| <a href="#">PMOVFL</a>   | Byte overflow                                           |
| <a href="#">RCVDSUCC</a> | Correctly received messages                             |
| <a href="#">WFACK</a>    | Wait-for-acknowledgement timeout                        |
| <a href="#">WFMSG</a>    | Wait-for-start-of-message timeouts                      |
| <a href="#">WFNR</a>     | Wait-for-idle message                                   |
| <a href="#">WFNX</a>     | Wait-for-link idle                                      |

**Registers for OM group PMMSGCNT**

| Register name            | Measures                       |
|--------------------------|--------------------------------|
| <a href="#">WFSND</a>    | Wait-for-send timeout          |
| <a href="#">XMITSUCC</a> | Correctly transmitted messages |

**DNACK****Register type**

Peg

**Description**

Double negative acknowledgements (DNACK)

Register DNACK counts double negative acknowledgement messages that the IUC link receives from the C-side XPM.

The C-side XPM sends double negative acknowledgement messages when the LCM tries again but fails to send a message.

This register holds the contents of LCM maintenance counter DNACK.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**IDLSTATE****Register type**

Peg

**Description**

Spurious frame interrupts (IDLSTATE)

Register IDLSTATE counts spurious frame interrupts that can occur. For example, spurious frame interrupts can occur when noise is on the line.

This register holds the contents of LCM maintenance counter IDL\_STATE.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**IINVBYTE****Register type**

Peg

**Description**

Invalid byte (IINVBYTE)

Register IINVBYTE counts negative acknowledgement messages that the IUC link receives. The IUC receives these messages when a message has a byte count that is not correct. This message transmits from one LCM unit to the mate unit of this LCM.

Each message includes message length. The byte count is not always correct. The count is not correct if an LCM unit receives a message that contains more bytes than the given message length.

This register holds the contents of LCM maintenance counter IUC\_INV\_D\_BYTE.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**IINVCHAR****Register type**

Peg

**Description**

Invalid character (IINVCHAR)

Register IINVCHAR counts negative acknowledgement messages that the IUC link receives. The IUC receives these messages when a



message contains characters that are not correct. This message transmits from one LCM unit to the mate unit of this LCM.

This register holds the contents of LCM maintenance counter IUC\_INVD\_CHAR.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**IINVCKSM****Register type**

Peg

**Description**

Invalid checksum (IINVCKSM)

Register IINVCKSM counts negative acknowledgement messages that the IUC link receives. The IUC receives these messages when a message has a checksum that is not correct. This message transmits from one LCM unit to the mate unit of this LCM.

A checksum in a message from one unit of the LCM is not correct if the checksum differs from the calculated checksum. The mate of this LCM calculates the checksum. The calculated checksum receives the message.

This register holds the contents of LCM maintenance counter IUC\_INVD\_CHKSUM.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**IINVMSG****Register type**

Peg

**Description**

Invalid message (IINVMSG)

Register IINVMSG counts negative acknowledgement messages that the IUC link receives. The IUC receives these messages when a message is not correct. This message transmits from one LCM unit to the mate unit of this LCM.

This register holds the contents of LCM maintenance counter IUC\_INVD\_MSG.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**INACK****Register type**

Peg

**Description**

Inter unit communication link negative acknowledgements (INACK)

Register INACK counts negative acknowledgement messages that the IUC link receives. The IUC receives these messages when a message from one unit of the LCM to the mate unit has one of the following:

- a character that is not correct
- a byte count that is not correct
- a checksum that is not correct
- a message that is not correct

This register holds the contents of LCM maintenance counter IUC\_LINK\_NACK.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**INVNODE****Register type**

Peg

**Description**

Invalid node (INVNODE)

Register INVNODE counts messages the LCM receives from the C-side XPM that have a node number that is not correct.

An LCM node number in a message is not always correct. The number is not correct if the number is not the node number assigned to the LCM that receives the message.

This register holds the contents of LCM maintenance counter INV\_NODE.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**NULLMSG****Register type**

Peg

**Description**

Null messages (NULLMSG)

Register NULLMSG counts null messages the IUC link receives from the C-side XPM that are not reset messages.

This register contains the contents of LCM maintenance counter NULL\_MSG\_RCVD.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMCRC****Register type**

Peg

**Description**

Not correct cyclic redundancy check (PMCRC)

Register PMCRC counts messages the LCM receives from the C-side XPM that have cyclic redundancy check (CRC).

The CRC in a message from the C-side XPM is not correct if the CRC differs from the calculated CRC. The LCM that receives the message calculates this CRC.

The LCM can send a negative acknowledgement in response to a CRC that is not correct. If the LCM sends this response, the C-side XPM attempts a second transmission of the message.

This register contains the contents of LCM maintenance counter CRC.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMNACK****Register type**

Peg

**Description**

Single negative acknowledgement (PMNACK)

Register PMNACK counts single negative acknowledgement messages that the IUC link receives from the C-side XPM. These negative acknowledgement messages indicate a problem with the reception of a message from the LCM.

The LCM attempts to transmit the message again.

This register contains the contents of LCM maintenance counter NACK.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMOVFL****Register type**

Peg

**Description**

Byte overflow (PMOVFL)

Register PMOVFL counts messages from the C-side XPM to the LCM that have more than the permitted number of bytes.

This register holds the contents of LCM maintenance counter OVFL.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RCVDSUCC****Register type**

Peg

**Description**

Correctly received messages (RCVDSUCC)

Register RCVDSUCC counts messages from the C-side XPM that the LCM receives.

This register holds the contents of LCM maintenance counter RCVD\_SUCC.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**WFAACK**

**Register type**

Peg

**Description**

Wait-for-acknowledgement timeout (WFAACK)

Register WFAACK counts wait-for-acknowledgement timeouts on messages from the LCM to the C-side XPM.

This register holds the contents of LCM maintenance counter WFAACK.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**WFMSG**

**Register type**

Peg

**Description**

Wait-for-start-of-message timeouts (WFMSG)

Register WFMSG counts wait-for-start-of-message timeouts on messages from the C-side XPM to the LCM.

This register holds the contents of LCM maintenance counter WFMSG.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**WFNR****Register type**

Peg

**Description**

Wait-for-idle message (WFNR)

Register WFNR counts wait-for-idle messages from C-side XPM to the LCM after a message transfer.

This register holds the contents of LCM maintenance counter WFNR.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**WFNX****Register type**

Peg

**Description**

Wait-for-link idle (WFNX)

Register WFNX counts wait-for-link idle messages received after a negative acknowledgement of a message transfer from the C-side XPM.

This register holds the contents of LCM maintenance counter WFNX.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**WFSND****Register type**

Peg

**Description**

Wait-for-send timeout (WFSND)

Register WFSND counts wait-for-send timeouts on messages from the LCM to the C-side XPM.

This register holds the contents of LCM maintenance counter WFSND.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**XMITSUCC****Register type**

Peg

**Description**

Correctly transmitted messages (XMITSUCC)

Register XMITSUCC counts messages transmitted from the LCM to the C-side XPM.

This register holds the contents of LCM maintenance counter XMIT\_SUCC.

**Associated registers**

There are no associated registers.



**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

---

## PMOVLD

---

### Description

Peripheral module overloaded (PMOVLD)

Register PMOVLD counts originations and terminations that the extended multiprocessor system (XMS)-based peripheral modules (XPM) denies. The line trunk controller (LTC), the line group controller (LGC), and digital trunk controller (DTC) are examples of XPMs. Register PMOVLD counts denied originations for the following international XPMs: international LGC (ILGC) and international DTC (IDTC).

The system denies originations in order to reduce the processing load of an overloaded peripheral module (PM). The system also denies originations to make sure that calls in progress complete without degradation of service. The system denies terminations if no originations that the system can deny are present.

An overload condition occurs when a PM resource for processing becomes exhausted. The following conditions can cause the PM to overload:

- a hardware failure on the peripheral side (P-side) of the overloaded PM
- a network hardware failure
- entry changes that result in extensive messaging to the PM
- an overconfigured PM

For lines, registers PORGDENY and PTRMDENY increase with the line concentrating module (LCM) that owns the line. For trunks, the same registers increase with the XPM that owns the trunk.

The system uses the data that PMOVLD supplies to monitor the performance of XPMs. The system also uses the data to determine if the XPMs are over configured.

The OM group PMOVLD provides one tuple for each XPM that is in-service (InSv) or in-service trouble (ISTb). The following table lists the key and info fields associated with OM group PMOVLD.

| Key field | Info field                                                                                                                                             |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| none      | PMOVLD_INFO_TYPE consists of the PM name (for example, LCM or LGC) and the internal PM number. The PM name is the name used to post the PM at the MAP. |

## Related functional groups

The following functional groups associate with the OM group PMOVLD:

- DMS-100 Local
- DMS-100/200 Local/Toll
- DMS-100/200 Local/Toll with TOPS
- DMS-200 Toll
- DMS-200 with TOPS
- DMS-100 Meridian
- DMS-MTX Mobile Telephone Exchange
- DMS-250 Toll/Tandem
- DMS-300 Gateway
- Meridian SL-100 PBX

## Registers

The following table lists the registers associated with OM group PMOVLD and what they measure. For a description of a register, click on the register name.

### Registers for OM group PMOVLD

| Register name            | Measures                       |
|--------------------------|--------------------------------|
| <a href="#">PORGDENY</a> | Peripheral origination denied  |
| <a href="#">PTRMDENY</a> | Peripheral terminations denied |

**PORGDENY****Register type**

Peg

**Description**

Peripheral origination denied (PORGDENY)

For North American XPMs, PORGDENY counts each line or trunk origination that an in-service PM denies. An in-service PM can deny a line or trunk origination because of an overload condition.

The overloaded PM does not process a denied trunk origination. To originate the call again, the caller must go on hook and try again. The overloaded PM does not give any treatment. A guaranteed dial tone (GDT) handles a denied line origination.

The XPM sets the internal overload indicator if the XPM increases PORGDENY for more than 2% of its originations. The system calculates this figure from a minimum sample size of 100 originations. The XPM also sets the internal overload indicator if the system delays a trunk origination by more than 4 s. When the overload indicator passes to the CC in a maximum of one minute, the following events occur

- a minor alarm sounds
- the XPM state changes to in-service trouble (ISTB)
- the system generates PM128

For North American XPMs, PORGDENY counts denied trunk originations for the line trunk controller, remote cluster controller, and digital trunk controller. Only trunk types that use winks need trunk overload control. An example of a wink is a multifrequency wink that signals when the system receives far-end off-hook signals. Register PORGDENY does not count trunk types that use immediate dial pulse after off-hook signals do not have overload protection.

For international XPMs (ILGC and IDTC), central control overload conditions guarantee a dial tone. Register PORGDENY does not increase for line originations. Register PORGDENY increases when the flow control queue is full and the system cannot guarantee a dial tone. Register PORGDENY counts either of the following two events

- an origination message remains on a flow control queue for a minimum of 3 s
- more than three origination messages from one terminal are on the flow control queue

**Associated registers**

Register CP\_ORIGDENY counts call originations that the central control denies.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM106 indicates when an XPM operates after the XPM was in an overload condition. Log PM106 also indicates when the PM is in service after the PM was in-service trouble.

For North American XPMs, PM106 also indicates the system no longer denies originations or terminations.

For international X series-2 PMs, PM106 indicates that a maximum of 2% of the last 1000 originations remain on the flow control queue for more than 3 s.

The system generates PM128 when a PM changes state from in service to in-service trouble because of an overload condition.

For North American XPMs, PM128 indicates that the PM is overloaded and that the system denies call originations or terminations. Refer to PM106 for required action.

For international XPMs, the system generates PM128 if a minimum of 2% of the last 1000 originations remain on the flow control queue. The system generates PM128 if a minimum of 2% of the last 1000 originations remain in the flow control queue for more than 3 s. Refer to PM106 for required action.

If the system generates logs PM106 and PM128, record the following information to help determine the cause of the overload condition:

- hardware failures on the switch
- manual action performed on the overloaded PM
- OMs that groups PMOVL and CP generated in the overload period
- data that relates to the overloaded PM and its peripheral-side (P-side) nodes

**PTRMDENY****Register type**

Peg

**Description**

Peripheral terminations denied (PTRMDENY)

Register PTRMDENY counts terminations that North American XPMs deny because of an overload condition. The PM must be in service before the overload condition occurs for PTRMDENY to count terminations.

A PM denies terminations when the incoming flow control queue reaches the upper limit and the PM cannot deny originations. After the PM denies a termination, the PM sends the central control a Problem message, and the system brings the call down.

Register PTRMDENY does not increase for international XPMs.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

Log PM106 indicates that an XPM operates after the XPM was in an overload condition. Log PM106 also indicates that the PM is in service after the PM was in-service trouble.

For North American XPMs, PM106 also indicates that the system does not deny originations or terminations.

For International XPMs, PM106 indicates that a maximum of 2% of the last 1000 originations remain. The last 1000 originations do not remain on the flow control queue for more than 3 s.

The system generates PM128 when a PM changes state from in-service to in-service trouble because of an overload condition.

For North American XPMs, PM128 indicates that the PM is overloaded and that the system denies call originations or terminations. Refer to PM106 for required action.

For international XPMs, the system generates PM128 if a minimum of 2% of the last 1000 originations remain on the flow control queue for more than 3 s. See PM106 for required action.

If the system generates logs PM106 and PM128, record the following information to help determine the cause of the overload condition:

- hardware failures on the switch
- manual action performed on the overloaded PM
- OMs groups PMOVL and CP generated in the overload period
- all data that relates to the overloaded PM and its peripheral side (P-side) nodes

## PMSTAT

### Description

Peripheral module status (PMSTAT)

The OM group PMSTAT records statistics for each unit of the extended line concentrating module (XLCM) family of peripheral modules (PM). This OM group records statistics on microprocessor occupancy. Occupancy is the amount of time in each 10 s period that a microprocessor performs work. This OM group records the following about the XLCM:

- overhead constant
- total processor occupancy
- call processing occupancy
- peak occupancy
- low occupancy
- available time of the XLCM

The OM group PMSTAT provides two tuples for each line concentrating module (LCM). The following table lists the key and info fields associated with OM group PMSTAT.

| Key field     | Info field |
|---------------|------------|
| PMSTAT_OM_KEY | none       |

### Related functional groups

The XLCM peripherals functional groups associate with OM group PMSTAT.

### Registers

The following table lists the registers associated with OM group PMSTAT and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PMSTAT

| Register name           | Measures                   |
|-------------------------|----------------------------|
| <a href="#">PMAVAIL</a> | PM available               |
| <a href="#">PMAVCP</a>  | PM average call processing |



**Registers for OM group PMSTAT**

| Register name            | Measures                            |
|--------------------------|-------------------------------------|
| <a href="#">PMAVOC</a>   | PM average occupancy                |
| <a href="#">PMLOWOC</a>  | PM low occupancy (low-water mark)   |
| <a href="#">PMOVHEAD</a> | PM overhead                         |
| <a href="#">PMPEAKOC</a> | PM peak occupancy (high-water mark) |

**PMAVAIL****Register type**

Usage

**Description**

PM available (PMAVAIL)

Register PMAVAIL updates every 10 s to record the average time the microprocessor does not have work to perform. This register records the time that the microprocessor does not have work to perform as a percentage. Register PMAVAIL records the average time in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval. The available time of the PM is inversely proportional to the average occupancy time of the PM (PMAVOC).

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMAVCP****Register type**

Usage

**Description**

PM average call processing (PMAVCP)

Register PMAVCP updates every 10 s to record the average processor occupancy (in percent) used for call processing. This register records the average processor occupancy as a percentage. Register PMAVCP records the average processor occupancy in 15 min intervals or 30 min

intervals. The data in table OFCENG determines the period of the interval.

To predict average occupancy, you must gather high-water marks for the busiest hours of the busiest days of the year. Follow the High-Day Busy Hour or the Extreme Value Engineering supply concept. Use this data to calculate and adjust supply of processor occupancy. Northern Telecom recommends that the average occupancy of LCM processors is not over 70%.

#### **Associated registers**

$\text{PMAVCP} = \text{PMAVOC} - \text{PMOVHEAD}$

Register PMAVOC updates every 10 s to record the average processor occupancy. This register records the average processor occupancy as a percentage. Register PMAVOC records the average processor occupancy in 15 min intervals and 30 min intervals. The data in table OFCENG determines the period of the interval.

Register PMOVHEAD

#### **Extension registers**

There are no extension registers.

#### **Associated logs**

There are no associated logs.

### **PMAVOC**

#### **Register type**

Usage

#### **Description**

PM average occupancy (PMAVOC)

Register PMAVOC updates every 10 s to record the average processor occupancy (in percent). This register records the average processor occupancy as a percentage. Register PMAVOC records the average processor occupancy in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval.

#### **Associated registers**

$\text{PMAVOC} = 100 - \text{PMAVAIL}$

Register PMAVAIL updates every 10 s to record average amount of time the microprocessor does not have work to perform. This register records the average amount of time as a percentage. Register PMAVOC records the average amount of time in 15 min intervals or 30

min intervals. The data in table OFCENG determines the period of the interval.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMLOWOC****Register type**

Usage

**Description**

PM low occupancy (low-water mark) (PMLOWOC)

Register PMLOWOC records the lowest processor occupancy value. This register records the lowest processor occupancy value as a percentage. This register records the lowest processor occupancy value in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval. The system takes samples every 10 s in each 15 min interval or 30 min interval. The system takes samples to determine the highest available time. The following equation calculates the low occupancy value:

$$\text{PMLOWOC} = 100 - \text{highest available time}$$

To predict lowest use accurately, gather low-water marks. Gather low-water marks for the least busy hours of the least busy days of the year. Use this data to calculate and adjust the supply of processor occupancy. Make sure the processor occupancy is not less than 20% use during the least busy times.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMOVHEAD****Register type**

Usage

**Description**

PM overhead (PMOVHEAD)

Register PMOVHEAD records the amount of processor use the system dedicates to overhead. This register records the amount of processor use the system dedicates to overhead in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval. Use the overhead value as a constant to calculate the average call processor use (PMAVCP).

Use the overhead constant over a 24-hr period. During this time, the system checks the available time value. This register checks the available time value in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval. The system determines if the available time value is the highest recorded value. (Note: highest availability = lowest occupancy.) If this value is higher than any of the previous records, the system stores the value. The system can use the value to obtain the overhead constant for the following 24 hr period. This method allows for a calculation of overhead. The calculation takes into account configuration changes and additional services allocated in the past 24 hr period.

**Associated registers**

Register PMOVHEAD = The lowest PMAVOC over 24 hr period

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PMPEAKOC****Register type**

Usage

**Description**

PM peak occupancy (high-water mark) (PMSTAT)

Register PMPEAKOC records peak processor occupancy as a percentage. This register records peak processor occupancy in 15 min intervals or 30 min intervals. The data in table OFCENG determines the period of the interval. The system takes samples every 10 s in each 15 min interval or 30 min interval. The system takes samples to determine the lowest available time. The following equation derives peak occupancy:

$$\text{PMPEAKOC} = 100 - \text{lowest available time}$$

Register PMPEAKOC registers the highest occupancy the system scans during the reporting period. Tasks of high activity and short duration cause register PMPEAKOC to report high values. An example of a task of high activity and short duration is internal system maintenance. Internal system maintenance runs continuously. At this time, current call processing is small or none. These high values can create a false belief that LCMs are near full capacity when the LCMs are in established engineering guidelines. Northern Telecom recommends that you must not use register PMPEAKOC to evaluate the supply and setup of LCM. Northern Telecom provides register PMAVCP for that purpose.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## PMTYP

### Description

Registers in OM group Peripheral Module Type (PMTYP) count peripheral module (PM) errors, faults, and state changes for all PMs of the same type (for example, all line group controllers).

Table PMEXCEPT must contain the node number of each PM excluded from group PMTYP totals, including commissioned modules and modules in test. If the office parameter OMINERLANGS in table OFCOPT is set to Y (yes), the output from the following usage registers is in deci-erlangs: PMTMSBU, PMTUSBU, PMTMMBU, PMTUMBU, PMTDRMBU, and PMTDRSBU.

The following table lists the key and field info associated with OM group PMTYP.

| Key field                                                                                                                                                                         | Info field                                                                                                                                                                                 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PM_TYPE<br>accesses the tuple. Table <a href="#">Field values and PM types for OM group PMTYP</a> lists the values for the key field and the PMs that correspond to these values. | PMTYP_OM_INFO_TYPE<br>includes the total number of PMs of the same type on the switch. The total contained in this field does not include PMs with node numbers entered in table PMEXCEPT. |

### Field values and PM types for OM group PMTYP (Sheet 1 of 7)

| Value | PM type                            |
|-------|------------------------------------|
| ADTC  | Austrian digital trunk controller  |
| ALCM  | Austrian line concentrating module |
| ALGC  | Austrian line group controller     |
| AP    | Application processor              |
| APU   | Application processing unit        |
| ARCC  | Austrian remote cluster controller |
| CFI   | Channel frame interface            |
| CFP   | Channel frame processor            |

**Field values and PM types for OM group PMTYP (Sheet 2 of 7)**

| <b>Value</b> | <b>PM type</b>                             |
|--------------|--------------------------------------------|
| CSC          | Cell site controller                       |
| STM          | Conference trunk module                    |
| DA           | Directory assistance database              |
| DCA          | Austrian digital carrier module            |
| DCM          | Digital carrier module                     |
| DCM250       | Digital carrier module DMS-250             |
| DES          | Digital echo suppressor                    |
| DFI          | Direct fiber interface                     |
| DLM          | Digital line module                        |
| DTC          | Digital trunk controller                   |
| DTC7         | CCS7 Digital trunk controller              |
| DTCI         | Digital trunk controller for ISDN          |
| DTCO         | Digital trunk controller offshore          |
| DTM          | Digital trunk module                       |
| EIU          | Ethernet interface unit                    |
| ELCM         | Enhanced line concentrating module         |
| ESA          | Emergency stand-alone                      |
| EXND         | External node                              |
| FRCC         | Force (download) remote cluster controller |
| FRIU         | Frame relay interface unit                 |
| FILP         | File processor                             |
| GIC          | Generic interface controller               |
| HFT          | HDLC frame transceiver                     |

**Field values and PM types for OM group PMTYP (Sheet 3 of 7)**

| <b>Value</b> | <b>PM type</b>                                   |
|--------------|--------------------------------------------------|
| HSI          | High speed interface                             |
| HSI2         | High speed interface series 2                    |
| HSIE         | High speed interface extended                    |
| IAC          | ISDN access controller                           |
| ICP          | Integrated cellular peripheral                   |
| ICRM         | Integrated cellular remote module                |
| IDT          | Integrated digital terminal                      |
| IDTC         | International digital trunk controller           |
| ILCM         | International line concentrating module          |
| LGC          | International line group controller              |
| LTC          | International line trunk controller              |
| XLCM         | International extended line concentrating module |
| IPE          | Intelligent peripheral equipment                 |
| ITAC         | International TATS access controller             |
| LCM          | Line concentrating module                        |
| LCME         | Enhanced line concentrating module               |
| LCMI         | ISDN line concentrating module                   |
| LCOM         | LIU-COM (link interface unit data communication) |
| LDT          | Line appearance on a digital trunk               |
| LGC          | Line group controller                            |
| LGCI         | Line group controller ISDN                       |
| LGCO         | Line group controller offshore                   |



**Field values and PM types for OM group PMTYP (Sheet 4 of 7)**

| <b>Value</b> | <b>PM type</b>                    |
|--------------|-----------------------------------|
| LIM          | Link interface module             |
| LIU          | Link interface unit               |
| LIU7         | CCS7 link interface unit          |
| HLIU         | High-speed link interface unit    |
| HSLR         | High-speed link router            |
| LM           | Line module                       |
| LRU          | Line resource unit                |
| LTC          | Line trunk controller             |
| LTCI         | Line trunk controller ISDN        |
| MMA          | Austrian maintenance trunk module |
| MSB6         | Message switch buffer for CCIS6   |
| MSB7         | Message switch buffer for CCIS7   |
| MTM          | Maintenance trunk module          |
| NIU          | Network interface unit            |
| OAU          | Office alarm unit                 |
| OPM          | Outside plant module              |
| ORDB         | Operator reference database       |
| PDTC         | PCM30 digital trunk controller    |
| PLGC         | PCM30 line group controller       |
| PND          | PNODE                             |
| PRCC         | PCM30 remote cluster controller   |
| PSP          | Programmable signal processor     |
| PTM          | Packaged trunk module             |

**Field values and PM types for OM group PMTYP (Sheet 5 of 7)**

| <b>Value</b> | <b>PM type</b>                        |
|--------------|---------------------------------------|
| RCC          | Remote cluster controller             |
| RCC2         | Compact remote cluster controller     |
| RCCI         | ISDN remote cluster controller        |
| RSCO2        | Remote switching center offshore 2    |
| RCS          | Remote concentrator SLC-96            |
| RCT          | Remote concentrator terminal          |
| RCU          | Remote carrier urban                  |
| RLC          | Remote line controller                |
| RLCM         | Remote line concentrating module      |
| RLM          | Remote line module                    |
| RMM          | Remote maintenance module             |
| RMSC         | Remote mobile switching center        |
| RSC          | Remote switching center               |
| RSCO         | Remote switching center offshore      |
| RSM          | Remote service module                 |
| SCM          | Subscriber carrier module             |
| SMA          | Subscriber module access              |
| SMR          | Subscriber carrier module-100 rural   |
| SMS          | Subscriber carrier module-100S        |
| SMSR         | Subscriber carrier module-100S remote |
| SMU          | Subscriber carrier module-100 urban   |
| SPM          | Service peripheral module             |
| SRCC         | SONET remote cluster controller       |

**Field values and PM types for OM group PMTYP (Sheet 6 of 7)**

| <b>Value</b> | <b>PM type</b>                    |
|--------------|-----------------------------------|
| SRU          | Small remote unit (ISDN LCM)      |
| STCM         | Signal terminal controller module |
| STM          | Service trunk module              |
| STS          | Standardized traffic statistics   |
| SVR7         | CCS7 server                       |
| T8A          | Trunk module for CCITT circuits   |
| TACC         | TATS access controller            |
| TAN          | Test access network               |
| TDTC         | MOC DTC (MOC is an NT licensee)   |
| TLGC         | MOC LGC (MOC is an NT licensee)   |
| TLTC         | MOC LTC (MOC is an NT licensee)   |
| TM           | Trunk module                      |
| TM2          | Trunk module—two wire             |
| TM4          | Trunk module—four wire            |
| TM8          | Trunk module ATT testing          |
| TMA          | Trunk module Austria              |
| TMS          | TOPS message switch               |
| TPC          | TOPS position controller          |
| TRCC         | MOC RCC (MOC is a NT licensee)    |
| VLCM         | Virtual line concentrating module |
| VPU          | Voice processing unit             |
| VSR          | Very small remote                 |
| VSROM        | Very small remote                 |

**Field values and PM types for OM group PMTYP (Sheet 7 of 7)**

| Value | PM type                                   |
|-------|-------------------------------------------|
| XLCM  | Expanded memory line concentrating module |
| XLIU  | X.25/X.75 link interface unit             |
| XRLCM | Extended remote line concentrating module |

**Related functional groups**

The following functional groups associate with OM group PMTYP:

- DMS-100 Local Office
- DMS-100/200 Combined Local/Toll office
- DMS-100/200 Combined Local/Toll Office with TOPS
- DMS-200 Toll Office
- DMS-200 with TOPS
- DMS-MTX Mobile Telephone Exchange
- DMS-250 Toll/Tandem Switch
- DMS-300 Gateway
- Meridian SL-100 PBX

**Registers**

The following table lists the registers associated with OM group PMTYP and what they measure. For a description of a register, click on the register name.

**Note:** The OM group PMTYP provides one tuple for each PM type.

**Registers for OM group PMTYP (Sheet 1 of 2)**

| Register name            | Measures                                   |
|--------------------------|--------------------------------------------|
| <a href="#">PMTCTDG</a>  | PM total circuit diagnostics run           |
| <a href="#">PMTCTFL</a>  | PM total circuit diagnostics failed        |
| <a href="#">PMTCTOP</a>  | PM total circuit diagnostics outside plant |
| <a href="#">PMTDRERR</a> | PM total drawer error                      |
| <a href="#">PMTDRFLT</a> | PM total drawer faults                     |

**Registers for OM group PMTYPE (Sheet 2 of 2)**

| Register name            | Measures                                           |
|--------------------------|----------------------------------------------------|
| <a href="#">PMTDRMBU</a> | PM total drawer manual busy use                    |
| <a href="#">PMTDRSBU</a> | PM drawer system busy use                          |
| <a href="#">PMTERR</a>   | PM total errors                                    |
| <a href="#">PMTFLT</a>   | PM total faults                                    |
| <a href="#">PMTINTEG</a> | PM total integrity failures                        |
| <a href="#">PMTMBP</a>   | PM total transitions to manual busy                |
| <a href="#">PMTMBTCO</a> | PM total manual busy terminals cut off             |
| <a href="#">PMTMCXFR</a> | PM total manual cold transfers                     |
| <a href="#">PMTMMBU</a>  | PM total module manual busy use                    |
| <a href="#">PMTMSBU</a>  | Peripheral module total module system busy use     |
| <a href="#">PMTMWXFR</a> | PM total manual warm transfers                     |
| <a href="#">PMTPSERR</a> | PM total peripheral side (P-side) errors           |
| <a href="#">PMTPSFLT</a> | PM total peripheral side (P-side) faults           |
| <a href="#">PMTRGERR</a> | PM total ringing generator errors while in service |
| <a href="#">PMTRGFLT</a> | PM total ringing generator faults while in service |
| <a href="#">PMTSBP</a>   | PM total transitions to system busy                |
| <a href="#">PMTSBTCO</a> | PM total system busy terminals cut off             |
| <a href="#">PMTSCXFR</a> | PM total system cold transfers                     |
| <a href="#">PMTSWXFR</a> | PM total system warm transfers                     |
| <a href="#">PMTUMBU</a>  | PM total unit manual busy use                      |
| <a href="#">PMTUSBU</a>  | PM total unit system busy use                      |

**PMTCCTDG****Register type**

Peg

**Description**

PMTCCTDG counts system-initiated tests run on line or trunk cards because of recurring problems during call processing. PMTCCTDG counts this type of maintenance event for a group of PMs of the same type. Maintenance conditions that cause PMTCCTDG to increase vary by PM type.

For digital carrier modules, PMTCCTDG counts tests run on a trunk interface card because of problems during call processing. The diagnostics determine if:

- a DS-1 interface card was removed
- loss of the systems ability to frame caused a local or remote-carrier group alarm state

For line modules and digital line modules, PMTCCTDG increases when system-initiated diagnostics are run on line cards.

For trunk modules, PMTCCTDG increases when the following diagnostics are run on any trunk interface card or service circuit:

- a check that the right card types are on the shelf
- test relay operation
- operation and release of signal distribution points and analysis of scan results
- a check for transmission loss in looparound mode

For extended multiprocessor system (XMS)-based peripheral modules (XPM), PMTCCTDG increases when a system-initiated diagnostic runs on lines or trunks. The diagnostic is run because recurring problems during call processing.

**Associated registers**

The following registers are associated with PMTCCTDG:

- [PMTCCTFL](#)
- PM\_PMTCCTDG counts system-initiated diagnostics for each PM.

**Extension registers**

None

**Associated logs**

PM110, TRK106

**PMTCCTFL****Register type**

Peg

**Description**

PMTCCTFL increases when system-initiated diagnostics determine the cause of a PM maintenance problem is one of the following:

- a card fault
- a missing or wrong card
- other fault conditions

PMTCCTFL counts these events for a group of PMs of the same type. The faults that cause the count of PMTCCTFL to increase are different for each PM.

For digital carrier modules, PMTCCTFL increases when:

- a card is removed
- a transmission error causes a carrier group alarm

For line modules, PMTCCTFL increases when diagnostics show:

- a fault in a PM, card, or facility
- a missing or wrong card

PMTCCTFL increases when diagnostics detect a wrong card, no card, or a bad card for:

- trunk modules
- digital carrier modules
- extended multiprocessor system (XMS)-based peripheral modules (XPM)

**Associated registers**

The following registers are associated with PMTCCTFL:

- [PMTCCTDG](#)
- PM\_PMCCTFL increases when a system-initiated diagnostic determines the cause of a PM maintenance problem by a fault condition. PM\_PMCCTFL increases for a separate PM.

**Extension registers**

None

**Associated logs**

PM109, PM183, TRK106

**PMTCCCTOP****Register type**

Peg

**Description**

PMTCCCTOP increases when system diagnostics detect a fault on a line or trunk circuit outside a switching office. The register counts the fault for a group of PMs of the same type.

Conditions that cause PMTCCCTOP to increase vary with PM types. The register increases only the first time the system detects a fault. PMTCCCTOP does not increase if the system detects the same fault when tests are run again.

For the digital carrier module and the trunk module, PMTCCCTOP increases when the signaling-test system at a switching office detects a fault. PMTCCCTOP detects a fault on a trunk circuit between the register and a far-end office. For example, PMTCCCTOP increases when an originating office does not receive a start-dial or wink signal from the far-end office in response to the off-hook signal the originating office sent.

For the line module, PMTCCCTOP increases when system diagnostics detect a fault on a line circuit outside the switching office.

For extended multiprocessor system (XMS)-based peripheral modules (XPM), PMTCCCTOP increases when system diagnostics detect a fault on a line or trunk outside the switching office.

**Associated registers**

PM\_PMCCTOP increases when system diagnostics detect a fault on a line or trunk circuit outside the switching office.

**Extension registers**

None

**Associated logs**

None



**PMTDRERR****Register type**

Peg

**Description**

PMTDRERR increases when an error in a line drawer causes the drawer to become in-service trouble. PMTDRFLT counts this type of fault for a group of PMs of the same type.

**Associated registers**

The following registers are associated with PMTDRERR:

- [PMTDRFLT](#)
- PM\_PMDRERR counts errors in a line drawer that cause the drawer to become in-service trouble.

**Extension registers**

None

**Associated logs**

PM102, PM181

**PMTDRFLT****Register type**

Peg

**Description**

PMTDRFLT counts the faults in a line drawer that cause the drawer to become system busy. PMTDRFLT counts this type of fault for a group of PMs of the same type.

**Associated registers**

The following registers are associated with PMTDRFLT:

- [PMTDRERR](#)
- PM\_PMDRFLT counts faults in a line drawer that cause the drawer to become system busy.

**Extension registers**

None

**Associated logs**

PM102, PM181

**PMTDRMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMTDRMBU records if the line drawers in a group of PMs of the same type are manual busy.

**Associated registers**

The following registers are associated with PMTDRMBU:

- [PMTDRSBU](#)
- PM\_PMDRMBU records if a line drawer in a PM is manual busy.

**Extension registers**

None

**Associated logs**

PM102, PM128

**PMTDRSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMTDRSBU records if the line drawers in a group of PMs of the same type are system busy.

**Associated registers**

The following registers are associated with PMTDRSBU:

- [PMTDRMBU](#)
- PM\_PMDRSBU records if a line drawer in the PM is system busy.

**Extension registers**

None

**Associated logs**

PM102, PM128

## PMTERR

### Register type

Peg

### Description

PMTERR counts errors in a group of PMs of the same type that are in service. The errors do not have to result in additional maintenance action for the system to count them.

For single-unit PMs (line, digital carrier, maintenance trunk, and trunk modules), PMTERR counts:

- command protocol violations
- RAM parity failures
- firmware errors
- controller message congestion
- test failures during the use of routine or initialization audits
- failure to respond to a message over either plane of a network

For two-unit XPMs (line concentrating modules, line group and line trunk controllers), PMTERR increases if one of the following maintenance events occur in either unit of the PM:

- errors that only result in the generation of a log
- errors resulting in more maintenance action
- integrity failures
- errors resulting in Who-Am-I messages
- changes in a unit from in service to central-side (C-side) or system busy
- restart reports
- any event that causes a fault and increases register PMTFLT

### Associated registers

The following registers are associated with PMTERR:

- [PMTFLT](#)
- PM\_PMERR counts errors on in-service PMs that have node numbers.
- PM2\_PM2ERR counts the same errors for PMs that do not have node numbers.

**Extension registers**

None

**Associated logs**

CCS231, CCS236, DDM101, DDM102, DDM104, DLC101, DPAC103, LOST108, LOST109, LOST111, MPC906, NET102, NPAC210, PM101, PM102, PM107, PM108, PM113, PM115, PM116, PM117, PM118, PM119, PM121, PM122, PM124, PM125, PM126, PM128, PM150, PM160, PM179, PM180, PM194, PM198, TRK123

**PMTFLT****Register type**

Peg

**Description**

PMFLT counts PM faults the system detects in a group of PMs of the same type that cause the entire PM or one unit of the PM to become system busy.

The register does not count the same fault in a following test when system diagnostics attempts to clear the fault. Conditions that cause PMTFLT to increase differ between single-unit PMs and XPMs.

For single unit PMs (line modules, digital carrier modules, and trunk modules), PMTFLT counts all errors that cause the PM to become system busy while the PM waits for either manual or system recovery.

For XPMs (line concentrating modules, line group and line trunk controllers), PMTFLT increases if:

- the system makes a PM or a single unit of a PM system busy.
- the system makes a C-side node or link manual busy and return it to service, which results in a change from C-side busy to system busy.

**Associated registers**

The following registers are associated with PMFLT:

- [PMTERR](#)
- PM\_PMFLT and PM2\_PM2FLT count faults that cause the system to make a PM or one unit of PM system busy. PM\_PMFLT counts faults for PMs that have node numbers. PM2\_PM2FLT counts the same faults for PMs without node numbers.

**Extension registers**

None

**Associated logs**

DLC102, DPAC104, MPC904, NPAC211, PM100, PM101, PM102, PM107, PM114, PM117, PM127, PM151, PM161, PM162, PM164, PM179, PM180, PM181, PM185, PM199

**PMTINTEG****Register type**

Peg

**Description**

PMTINTEG increases when the PM detects an integrity failure and reports it to the central control (CC). PMTINTEG counts integrity failures for a group of PMs of the same type.

**Associated registers**

PM\_PMTINTEG increases when the PM reports an integrity failure to the CC.

**Extension registers**

None

**Associated logs**

NET101, NET102, PM108, PM113, PM118, PM119, PM122, PM124, PM180, PM181, PM185, TRK122

**PMTMBP****Register type**

Peg

**Description**

PMTMBP increases when an in-service or in-service trouble PM becomes manual busy. The register counts the state change for a group of PMs of the same type. For line modules, PMTMBP increases when the LM becomes manual busy during manual warm and cold takeovers.

**Associated registers**

The following registers are associated with PMTMBP:

- [PMTSBP](#)
- PM\_PMTMBP increases when an in-service or in-service trouble PM becomes manual busy.

**Extension registers**

None

**Associated logs**

PM182, PM191

**PMTMBTCO****Register type**

Peg

**Description**

PMTMBTCO counts calls (terminals) cut off when a PM becomes manual busy. The register counts call-processing busy (CPB) and call-processing deloading (CPD) calls that are cut off for a group of PMs of the same type. Conditions that cause PMTMBTCO to increase vary by PM.

For digital carrier, line, and trunk modules, PMTMBTCO counts the subscriber calls cut off. Calls are cut off when the PM changes to manual busy from in service or in-service trouble.

A warm takeover can occur after the line module becomes manual busy. For line modules, subscriber calls in the talking state are not cut off. If a takeover does not occur, PMTMBTCO increases once for each subscriber call the system cuts off. If warm takeback occurs after the LM becomes manual busy, PMTMBTCO increases once for each subscriber call the system cuts off.

For extended multiprocessor system (XMS)-based peripheral modules (XPM), PMTMBTCO counts the subscriber calls cut off when the PM becomes manual busy. Subscriber calls must be call-processing busy or call-processing deloading for the PMTMBTCO to count the calls. PMTMBTCO increases once when the systems cuts off a call in the talking state.

**Associated registers**

The following registers are associated with PMTMBTCO:

- [PMTSBTCO](#)
- PM\_PMMBTCO and PM2\_PM2MBTCO count the subscriber calls (terminals) cut off when a PM becomes manual busy.

**Extension registers**

None

**Associated logs**

None

**PMTMCXFR****Register type**

Peg

**Description**

PMTMCXFR increases when a manual action causes an extended multiprocessor system (XMS)-based peripheral module (XPM) to perform a cold switch of activity (SWACT). PMTMCXFR counts manually initiated cold SWACTS for a group of PMs of the same type.

Examples of manual actions that can trigger a cold SWACT are:

- the execution of the SWACT command at the MAP terminal
- a manual request that sets the active unit manual busy while the inactive unit is in service

**Associated registers**

The following registers are associated with PMTMCXFR:

- [PMTSCXFR](#)
- PM\_PMMCXFR and PM2\_PM2MCXFR count manually initiated cold SWACTS for separate PMs. PM\_PMMCXFR counts manually initiated cold SWACTS for PMs that have node numbers. PM2\_PM2MCXFR counts manually initiated cold SWACTS for PMs that do not have node numbers.

**Extension registers**

None

**Associated logs**

PM128, PM180

**PMTMMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMTMMBU records if PMs of the same type are manual busy.

**Associated registers**

The following registers are associated with PMTMMBU:

- [PMTUMBU](#)
- PM\_PMMMBU and PM2\_PM2MMBU record if a separate PM is manual busy. PM\_PMMMBU provides a use count for PMs that have node numbers. PM2\_PM2MMBU provides a use count for PMs without node numbers.

**Extension registers**

None

**Associated logs**

CCS218, CCS233, PM105, PM128, PM170, PM182, PM191

**PMTMSBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMTMSBU records if PMs in an identical group are system busy. Hardware or software problems that cause the PM to become system busy vary with the PM type.

For digital carrier or trunk modules, the PM can become system busy because:

- the DCM or TM fails a routine audit
- message paths are not available to the DCM or TM
- The DCM or TM sends more than 200 non-requested trouble reports in one 10-minute audit period

For line modules, the PM can become system busy because:

- the system cannot reach the LM
- the control section of the LM did not pass an audit
- the LM reported more than 200 controller or line errors between audits



**Associated registers**

The following registers are associated with PMTMSBU:

- [PMTUSBU](#)
- PM\_PMMSBU and PM2\_PM2MSBU record if a separate PM is system busy. PM\_PMMSBU provides use counts for PMs that have node numbers. PM2\_PM2MSBU provides use counts for PMs without node numbers.

**Extension registers**

None

**Associated logs**

CCS234, PM102, PM128, PM170, PM183, PM190, PM192,

**PMTMWXFR****Register type**

Peg

**Description**

PMTMWXFR increases when manual maintenance forces a two-unit PM to switch activity (SWACT) or a take over a unit. PMTMWXFR counts this type of activity transfer for a group of PMs of the same type. The type of PM that the manual request acts on determines the activity transfer that occurs. PMTMWXFR increases if:

- a manual request forces an XPM (such as a line group controller or a line trunk controller) to perform a warm SWACT.
- a manual request forces one unit of a line concentrating module (LCM) to take over the other unit.

The system can force an LCM to perform a takeover. Takeover occurs when the system makes one unit of the LCM manual busy while the mate unit is in service. A takeback of activity does not increase PMTMWXFR.

Examples of manual actions that force an XPM to perform a warm SWACT are:

- the execution of the SWACT command at the MAP (maintenance and administration position) terminal
- the active unit of an XPM is made manual busy when the inactive unit is in service

**Associated registers**

The following registers are associated with PMTMWXFR:

- [PMTSWXFR](#)
- PM\_PMMWXFR and PM2\_PM2MWXFR count activity transfers for separate PMs. PM\_PMMWXFR counts transfers for PMs that have node numbers. PM2\_PM2MWXFR counts transfers for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM128, PM180

**PMTPSERR****Register type**

Peg

**Description**

PMTPSERR counts errors on:

- the P-side interface of an XMS-based XPM
- a link interface module (LIM) frame transport bus (F-bus) for a group of PMs of the same type

PMTPSERR increases with:

- line, trunk or link errors
- errors in interface cards that terminate lines, trunks, or links
- F-bus errors

**Associated registers**

The following registers are associated with PMTPSERR:

- [PMTPSFLT](#)
- PMPSEERR counts errors on the P-side interface of an XPM or an LIM F-bus.
- PM\_PMPSEERR counts errors for PMs that have node numbers.
- PM2\_PM2PSEERR counts errors for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM110

**PMTPSFLT****Register type**

Peg

**Description**

PMTPSFLT counts faults on the P-side interface of an XMS-based XPM or on the LIM F-bus for a group of PMs of the same type. The faults affect service and require more maintenance action.

PMTPSFLT increases when faults occur in:

- P-side interface cards that terminate trunks, lines, or links
- lines, trunks, and links serviced by the interface cards
- the F-bus

**Associated registers**

The following registers are associated with PMTPSFLT:

- [PMTPSERR](#)
- PM\_PMPSTFLT and PM2\_PMPSTFLT count faults on the P-side interface of an XPM or faults on the LIM F-bus. PM\_PMPSTFLT counts faults on the P-side interface of PMs that have node numbers. PM2\_PMPSTFLT counts faults for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM109, PM181, PM183

**PMTRGERR****Register type**

Peg

**Description**

PMTRGERR counts errors in ringing generators. The generators provide ringing and automatic number identification (ANI) coin functions to line concentrating modules or very small remotes (VSR). PMTRGERR counts the errors for a group of PMs of the same type.

PMTRGERR increases if the ringing generator is in or out of service at the time of the error. The LCM or VSR must be in service at the time of

the error. A single ringing generator can service both LCMs in the same frame. The system can count one ringing generator error four times, one time for each of the two line concentrating arrays in each of the two LCMs.

**Associated registers**

The following registers are associated with PMTRGERR:

- [PMTRGFLT](#)
- PM\_PMRGERR and PM2\_PM2RGERR count ringing generator errors for separate PMs.

**Extension registers**

None

**Associated logs**

PM160

**PMTRGFLT****Register type**

Peg

**Description**

PMTRGFLT counts service-affecting faults in ringing generators. The generators provide ringing and automatic number identification (ANI) coin functions to line concentrating modules (LCM) and to very small remotes (VSR) group of PMs of the same type. The ringing generator must be in service for PMTRGFLT to increase.

**Associated registers**

The following registers are associated with PMTRGFLT:

- PM\_PMRGFLT and PM2\_PM2RGFLT count ringing generator faults for separate PMs.
- PM\_PMRGFLT counts faults for PMs that have node numbers.
- PM2\_PM2RGFLT counts faults for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM161, PM162, PM163

**PMTSBP****Register type**

Peg

**Description**

PMTSBP increases when an in-service or in-service trouble PM module becomes system busy. The register counts the state change for a group of PMs of the same type.

The PM normally changes to central side (C-side) busy before the PM becomes system busy. If the PM correctly returns to service from C-side busy before the PM becomes system busy, PMTSBP does not increase. For line modules, PMTSBP increases when the LM becomes system busy during warm or cold takeovers.

**Associated registers**

The following registers are associated with PMTSBP:

- [PMTMBP](#)
- PM\_PMTSBP increases when a separate PM becomes system busy from an in-service or in-service trouble state.

**Extension registers**

None

**Associated logs**

DLC102, DPAC104, MPC904, NPAC211, PM107, PM183, PM190, PM192

**PMTSBTCO****Register type**

Peg

**Description**

PMTSBTCO counts subscriber calls (terminals) cut off when a PM becomes system busy. The register counts subscriber calls that the system drops for a group of PMs of the same type. Conditions that cause PMTSBTCO to increase vary by PM type.

For digital carrier module and trunk modules, PMTSBTCO counts subscriber calls that are cut when the state of an in-service or in-service trouble PM changes to C-side busy. Subscriber calls must be call-processing busy or call-processing deloading for PMTSBTCO to increase. C-side busy is an intermediate state that occurs before the PM becomes system busy.

For the line module (LM), PMTSBTCO counts subscriber calls that are cut when the line module becomes system busy. The calls must be call-processing busy or call-processing deloading for PMTSBTCO to increase.

If the line module recovers from C-side busy before it becomes system busy and the mate line module becomes system busy, associated register PMTSBTCO increases. PMTSBTCO increases by the number of subscriber calls the system busy mate drops. This increase occurs because the LM that performs the cold takeover is responsible for the calls of the mate LM. The LM cannot preserve these calls through the takeover.

If a warm takeover occurs when an LM becomes system busy, calls are not cut off and PMTSBTCO does not increase. An LM can perform a warm takeback of control of the line drawers in the LM after the system returns the LM to service from system busy. PMTSBTCO increases by the number of calls that the original system busy state change drops.

For XMS-based XPMs, PMTSBTCO counts subscriber calls cut when the PM becomes system busy. The subscriber calls must be call-processing busy or call-processing deloading for PMTSBTCO to increase. PMTSBTCO increases one time when the system drops a call in the talking state.

**Associated registers**

The following registers are associated with PMTSBTCO:

- [PMTMBTCO](#)
- PM\_PMSBTCO and PM2\_PM2SBTCO count calls that are cut when a separate PM becomes system busy.

**Extension registers**

None

**Associated logs**

None

**PMTSCXFR****Register type**

Peg

**Description**

PMTSCXFR increases when a system action causes an XMS-based XPM to perform a cold SWACT for a group of PMs of the same type. A cold SWACT occurs when the system:

- forces an XPM to perform a cold SWACT
- makes the active unit of an XPM system busy
- makes the central-side (C-side) links to the active unit of an XPM system busy

**Associated registers**

The following registers are associated with PMTSCXFR:

- [PMTMCXFR](#)
- PM\_PMSCXFR and PM2\_PM2SCXFR count system-initiated cold SWACTS for separate PMs.
- PM\_PMSCXFR counts cold SWACTS for PMs that have node numbers. PM2\_PM2SCXFR counts the same SWACTS for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM128, PM179, PM180, PM181

**PMTSWXFR****Register type**

Peg

**Description**

PMTSWXFR increases when system maintenance forces a two-unit PM to perform a warm SWACT or a unit takeover for a group of PMs of the same type. The activity transfer that the system performs varies by PM type. PMTSWXFR increases when:

- a system request forces an XMS-based XPM (such as a line controller or line group controller) to perform a warm SWACT.
- a system request forces one unit of a line concentrating module (LCM) to take over the other. Takeback of activity in the LCM does not increase PMTSWXFR.

A warm SWACT occurs when:

- the system forces an XPM to perform a warm SWACT
- the active unit of an XPM becomes system busy
- the C-side links to the active unit of an XPM become busy

An LCM can perform a takeover when:

- the system makes one unit of the LCM system busy while the mate unit is in service
- the system makes the C-side links to either LCM unit busy while the mate unit is in service

**Associated registers**

The following registers are associated with PMTSWXFR:

- [PMTMWXFR](#)
- PM\_PMSWXFR and PM2\_PM2SWXFR count activity transfers for separate PMs. PM\_PMSWXFR counts activity transfers for PMs that have node numbers. PM2\_PM2SWXFR counts transfers for PMs without node numbers.

**Extension registers**

None

**Associated logs**

PM128, PM179, PM180, PM181

**PMTUMBU****Register type**

Usage

**Scan rate**

100 seconds

**Description**

PMTUMBU records if PMs in a group of the same type are manual busy. The system determines if both units of an XMS-based XPM are manual busy. The register increases twice after each scan interval, once for each unit.

**Associated registers**

The following registers are associated with PMTUMBU:

- [PMTMMBU](#)
- PM\_PMUMBU and PM2\_PM2UMBU record if a separate PM is manual busy. PM\_PMUMBU provides a count for PMs that have node numbers. PM2\_PM2UMBU provides the count for PMs without node numbers.

**Extension registers**

None

**Associated logs**

CCS218, CCS233, PM105, PM128, PM182, PM191

**PMTUSB****Register type**

Usage



**Scan rate**

100 seconds

**Description**

PMTUSBU records if units of a group of identical PMs are system busy. If both units of an XMS-based XPM are system busy, PMTUSBU increases twice. The register increases after each scan interval, one time for each unit. For single-unit PMs such as line, digital carrier, and trunk modules, PMTUSBU increases once. Events that can cause one unit of an XPM to become system busy are:

- diagnostic failures
- excessive unsolicited messages
- auto unit resets

**Associated registers**

The following registers are associated with PMTUSBU:

- [PMTMSBU](#)
- PM\_PMUSBU and PM2\_PM2USBU record if a separate PM is system busy. PM\_PMUSBU provides a count for PMs that have node numbers. PM2\_PM2USBU provides counts for PMs without node numbers.

**Extension registers**

None

**Associated logs**

CCS234, PM102, PM128, PM170, PM183, PM190, PM192

## Ports

### Description

This OM group provides.

The following table lists the key and info fields associated with OM group Ports.

| Key field | Info field |
|-----------|------------|
|           |            |

### Related functional groups

The following functional groups are related to OM group Ports:

- RTP Media Portal

The following table lists the registers associated with OM group Ports and what they measure. For a description of a register, click on the register name.

#### Registers for OM group Ports

| Register name                         | Measures                                                 |
|---------------------------------------|----------------------------------------------------------|
| <a href="#">rtpMpPortUsage</a>        | RTP Media Portal port usage                              |
| <a href="#">connMapEntriesLWM</a>     | Lowest number of simultaneous ports detected.            |
| <a href="#">connMapEntriesHWM</a>     | Highest number of simultaneous ports detected.           |
| <a href="#">connMapEntries</a>        | Connections map entries.                                 |
| <a href="#">portUsageSampleCnt</a>    | Number of samples taken during this collection interval. |
| <a href="#">sampledRtpMpPortUsage</a> | Sums in-use ports every 100 seconds.                     |

**rtpMpPortUsage**  
**Register type**  
Usage

**Description**

Meter showing number of ports in use.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**connMapEntriesLWM****Register type**

Usage

**Description**

Lowest number of simultaneous ports detected to be in use during this collection interval.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**connMapEntriesHWM****Register type**

Usage

**Description**

Highest number of simultaneous ports detected to be in use during this collection interval.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**connMapEntries****Register type**

Usage

**Description**

Meter showing the number of connections in the host CPU's connection map.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**portUsageSampleCnt****Register type**

Counter

**Description**

The number of 100 second samples taken during this collection interval contributing to the averaged report.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**sampledRtpMpPortUsage****Register type**

Usage

**Description**

Provides the summing of the in use ports every 100 seconds.

**Associated registers**

None

**Extension registers**

None

**Associated logs**  
None

## PPCO

### Description

Pre Paid Coin Overtime (PPCO) measures certain Pre-Paid Coin Overtime Statistics.

OM group PPCO provides one tuple. The following table lists the key and info fields associated with OM group PPCO.

| Key field | Info field |
|-----------|------------|
| none      | none       |

### Related functional groups

Functional group ENSV Pre Paid Coin (ENSV0001) is associated with OM group PPCO:

### Registers

The following table lists the registers associated with OM group PPCO and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PPCO

| Register name            | Measures                           |
|--------------------------|------------------------------------|
| <a href="#">PPCOINI</a>  | Pre-Paid Coin Overtime Initial     |
| <a href="#">PPCOOVT</a>  | Pre-Paid Coin Overtime Overtime    |
| <a href="#">PPCODISC</a> | Pre-Paid Coin Overtime Disconnects |

#### PPCOINI

##### Register type

Peg

##### Description

Pre-Paid Coin Overtime Initial

PPCOINI is incremented when a sent-paid call uses PPCO. It keeps track of how many coin calls are made using trunks which are datafilled as pre-pay.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PPCOOVT****Register type**

Peg

**Description**

Pre-Paid Coin Overtime Overtime

PPCOOVT is incremented when the caller pays for their first overtime period. It keeps track of the amount of users who pay for at least the first overtime period.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PPCODISC****Register type**

Peg

**Description**

Pre-Paid Coin Overtime Disconnects

PPCODISC keeps track of disconnects which occur when a caller fails to deposit any funds for the next overtime period and after the necessary ACTS prompts and time-outs have occurred.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



---

## PRADCHL2

---

### Description

PRA D-channel layer 2 performance summary (PRADCHL2)

The operational measurements (OM) group PRADCHL2 monitors the layer 2 (Q.921) traffic that travels over the primary rate access (PRA) D channels in the integrated services digital network (ISDN) peripherals. Examples of ISDN peripherals are ISDN digital trunk controller (DTCI), line trunk controller (LTC), and Spectrum Peripheral Module (SPM).

The system increases the OMs in the ISDN digital trunk controller. The system collects OMs from the peripheral. This event occurs before the OM transfer from the active to the holding registers.

The PRADCHL2 peg registers record the following 1-minute collections:

- discarded transmit frames
- received frames with cyclic redundancy check (CRC) error
- received frames discarded that other errors cause
- correctly transmitted-service access point identifier (SAPI) 0 frames
- correctly received SAPI 0 frames
- link resets the ISDN signaling processor (ISP) causes
- link resets the far end device causes
- receiver not ready (RNR) frames the ISP to the far-end device transmits
- RNR frames received from the far-end device signaling channel (SIGL)
- reject (REJ) frames the ISP transmits
- REJ frames received from the far-end device
- PRA Q.931 messages the PRA flow control system discards

The OM group PRADCHL2 provides one tuple for each D channel. The PRADCHL2 peg registers record the following 15-minute collections:

- layer 2 service disruptions
- layer 3 service disruptions

The following table lists the key and info fields associated with OM group PRADCHL2.

| Key field                                                           | Info field                                                        |
|---------------------------------------------------------------------|-------------------------------------------------------------------|
| EXTERNAL_DCH-CKT identifies a single, primary, or backup D channel. | L2_OMINFO is the DTCl/LTCl number, circuit number, and time slot. |

## Related functional groups

Operating group ISDN associates with OM group PRADCHL2.

## Registers

The following table lists the registers associated with OM group PRADCHL2 and what they measure. For a description of a register, click on the register name.

### Registers for OM group PRADCHL2

| Register name            | Measures                                                              |
|--------------------------|-----------------------------------------------------------------------|
| <a href="#">PRDCRC</a>   | Cyclic redundancy check (CRC) errors                                  |
| <a href="#">PRDDISCR</a> | Received frames discarded                                             |
| <a href="#">PRDDISCT</a> | Transmit frames discarded                                             |
| <a href="#">PRDREJRX</a> | Reject (REJ) frames received                                          |
| <a href="#">PRDREJTX</a> | Reject frames (REJ) transmitted                                       |
| <a href="#">PRDRNRRX</a> | Receiver not ready (RNR) frames received                              |
| <a href="#">PRDRNRTX</a> | Receiver not ready (RNR) frames transmitted                           |
| <a href="#">PRDS0RX</a>  | Correctly received service access point identifier (SAPI) 0 frames    |
| <a href="#">PRDS0TX</a>  | Correctly transmitted service access point identifier (SAPI) 0 frames |
| <a href="#">PRDSBMRX</a> | Link resets, far-end device                                           |
| <a href="#">PRDSBMTX</a> | Link resets, ISDN signaling processor                                 |
| <a href="#">PRFLSHED</a> | Primary rate access (PRA) flow shed                                   |

**Registers for OM group PRADCHL2**

| Register name            | Measures                                  |
|--------------------------|-------------------------------------------|
| <a href="#">PRDL2SVD</a> | PRI D-channel layer 2 service disruptions |
| <a href="#">PRDL3SVD</a> | PRI D-channel layer 3 service disruptions |

**PRDCRC****Register type**

Peg

**Description**

Cyclic redundancy check (CRC) errors

Register PRDCRC is the 1-minute collection of the number of frames with CRC errors the system receives.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDDISCR****Register type**

Peg

**Description**

Received frames discarded (PRDDISCR)

Register PRDDISCR is the 1-minute collection of the number frames the system receives frames that the system discards because of errors other than CRC errors.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDDISCT****Register type**

Peg

**Description**

Transmit frames discarded (PRDDISCT)

Register PRDDISCT is the 1-minute collection of the number of transmit frames that the system discards.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDREJRX****Register type**

Peg

**Description**

Reject (REJ) frames received (PRDREJRX)

Register PRDREJRX is the 1-minute collection of the number of REJ frames the system receives from the far end.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDREJTX****Register type**

Peg

**Description**

Reject frames (REJ) transmitted (PRDREJTX)

Register PRDREJTX is the 1-minute collection of the number of REJ frames the ISDN signaling processor transmits.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDRNRRX****Register type**

Peg

**Description**

Receiver not ready (RNR) frames received (PRDRNRRX)

Register PRDRNRRX is the 1-minute collection of the number of frames the system from the far-end device receives.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDRNRTX****Register type**

Peg

**Description**

Receiver not ready (RNR) frames transmitted (PRDRNRTX)

Register PRDRNRTX is the 1-minute collection of the number of RNR frames the ISDN signaling processor transmits.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDSORX****Register type**

Peg

**Description**

Correctly received service access point identifier (SAPI) 0 frames (PRDSORX)

Register PRDSORX is the 1-minute collection of the number of SAPI 0 frames the system correctly receives.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDSOTX****Register type**

Peg

**Description**

Correctly transmitted service access point identifier (SAPI) 0 frames (PRDSOTX)

Register PRDSOTX is the 1-minute collection of the number of SAPI 0 frames that transmit correctly.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDSBMRX****Register type**

Peg

**Description**

Link resets, far-end device (PRDSBMRX)

Register PRDSBMRX is the 1-minute collection of the number of link resets the far-end device causes.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDSBMTX****Register type**

Peg

**Description**

Link resets, ISDN signaling processor (PRDSBMRX)

Register PRDSBMRX is the 1-minute collection of the number of link resets the ISDN signaling processor causes.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRFLSHED****Register type**

Peg

**Description**

Primary rate access (PRA) flow shed (PRFLSHED)

Register PRFLSHED is the 1-minute collection of the number of PRA Q.931 messages the PRA flow control system discards. The system discards a PRA Q.931 message when PRA overload controls are turned on. The system also discards a message when the peripheral

module has excessive numbers of messages that queue in the PRA flow control system.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**PRDL2SVD****Register type**

Peg

**Description**

PRI D-channel layer 2 service disruptions

Register PRDL2SVD is the 15 min collection of the number of PRA Q.921 layer 2 service disruptions that occur on PRI interfaces. This register monitors the following error conditions:

- link resets occurring on layer 2
- overflow of received frame buffer area

The system detects these disruptions at the ISDN signaling processor and reports the disruptions to the computing module (CM). The counts are valid only for Northern Telecom National ISDN (NTNI) PRA D channels.

Register PRDL2SVD is 16-bit. When this register reaches its capacity, the register remains pegged until the system resets the register. The system resets the register at 00:00 and 00:30 minutes every hour.

**Associated registers**

Register PRDL2SVD is the sum of the link reset occurrences on layer 2 and the overflow of the received frame buffer area. The existing registers, PRDSBMTX and PRDSBMRX, provide the link resets associated with the ISDN signal processor and the far-end device, respectively.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



**PRDL3SVD****Register type**

Peg

**Description**

PRI D-channel layer 3 service disruptions

Register PRDL3SVD is the 15 min collection of the number of PRA Q.931 layer 3 service disruptions that occur on PRI interfaces. This register monitors the following error conditions:

receipt of messages with invalid protocol discriminator

receipt of messages less than three octets in length

receipt of SETUP messages with call reference flag incorrectly set to 1

receipt of SETUP messages with missing or invalid mandatory information elements (IE)

receipt of messages other than SETUP messages containing an unallocated call reference value

These error conditions are Q.931 messaging errors that are not associated with the call. The system detects these disruptions at the uniform processor (UP) and reports the disruptions to the CM. The counts are valid only for NTNI PRA D-channels.

Register [PRDL2SVD](#) is 16-bit. When this register reaches its capacity, the register remains pegged until the system resets the register. The system resets the register at 00:00 and 00:30 minutes every hour.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## PRAFAC

### Description

Primary rate access facility

Primary rate access facility (PRAFAC) measures message traffic that is generated by network ring again (NRAG) on primary rate access (PRA) D channels. NRAG on PRA uses connectionless signaling on PRA, that is, no call is present.

Message traffic that is measured in PRAFAC includes origination, termination, and tandem messages. Counts are made for facility messages that are used to transfer high-layer protocols and for facility reject messages, which are sent when a facility message cannot be routed.

PRAFAC data can help identify network problems by measuring facility and facility reject messages from switch to switch.

OM group PRAFAC provides one tuple for each PRA trunk group. The following table lists the key and info fields associated with OM group PRAFAC.

| Key field                                     | Info field |
|-----------------------------------------------|------------|
| COMMON_LANGUAGE_NAME<br>identifies trunk CLLI | none       |

### Related functional groups

The ISDN functional group is associated with OM group PRAFAC.

### Registers

The following table lists the registers associated with OM group PRAFAC and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PRAFAC

| Register name            | Measures                                                  |
|--------------------------|-----------------------------------------------------------|
| <a href="#">DISCNGST</a> | Facility messages discarded due to switch congestion      |
| <a href="#">DISNORTX</a> | Facility messages discarded due to no routing translation |

**Registers for OM group PRAFAC**

| Register name            | Measures                                                          |
|--------------------------|-------------------------------------------------------------------|
| <a href="#">DISRTUNA</a> | Discarded facility messages due to route unavailable              |
| <a href="#">FACMSGOR</a> | Facility messages originated                                      |
| <a href="#">FACMSGTM</a> | Facility messages terminated                                      |
| <a href="#">FACMSGTR</a> | Facility messages transmitted                                     |
| <a href="#">REJCNGST</a> | Facility reject messages originated due to switch congestion      |
| <a href="#">REJMSGDS</a> | Facility reject messages discarded                                |
| <a href="#">REJMSGOR</a> | Facility reject messages originated                               |
| <a href="#">REJMSGTM</a> | Facility reject messages terminated                               |
| <a href="#">REJMSGTR</a> | Facility reject messages transmitted                              |
| <a href="#">REJNORTX</a> | Facility reject messages originated due to no routing translation |
| <a href="#">REJRTUNA</a> | Facility reject messages originated due to route unavailable      |

**DISCNGST****Register type**

Peg

**Description**

Facility messages discarded due to switch congestion

Facility messages discarded due to switch congestion (DISCNGST) counts facility messages that are discarded because of congestion in the DMS.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DISNORTX****Register type**

Peg

**Description**

Facility messages discarded due to no routing translation

Facility messages discarded due to no routing translation (DISNORTX) counts facility messages that are discarded because no route list was found in table MSGRTE.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DISRTUNA****Register type**

Peg

**Description**

Discarded facility messages due to route unavailable

Discarded facility messages due to route unavailable (DISRTUNA) counts facility messages that are discarded because the D channel route is not available.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**FACMSGOR****Register type**

Peg

**Description**

Facility messages originated

Facility messages originated (FACMSGOR) counts facility messages that are created and sent on a primary rate access (PRA) D channel.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**FACMSGTM****Register type**

Peg

**Description**

Facility messages terminated

Facility messages terminated (FACMSGTM) counts received facility messages that terminate at the DMS.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**FACMSGTR****Register type**

Peg

**Description**

Facility messages transmitted

Facility messages transmitted (FACMSGTR) counts facility messages that are received at a tandem switch and are transmitted to another switch. Both incoming and outgoing trunk groups are incremented.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJCNGST****Register type**

Peg

**Description**

Facility reject messages originated due to switch congestion

Facility reject messages originated due to switch congestion (REJCNGST) counts facility reject messages that originate at the DMS because of congestion at the DMS. Facility reject messages are sent back to the originators whenever it is not possible to route that facility message.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJMSGDS****Register type**

Peg

**Description**

Facility reject messages discarded

Facility reject messages discarded (REJMSGDS) counts facility reject messages that are discarded by the DMS whenever the message cannot be routed.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJMSGOR****Register type**

Peg

**Description**

Facility reject messages originated

Facility reject messages originated (REJMSGOR) counts facility reject messages that are created and sent on a primary rate access (PRA) D channel.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJMSGTM****Register type**

Peg

**Description**

Facility reject messages terminated

Facility reject messages terminated (REJMSGTM) counts received facility reject messages that terminate at the DMS.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJMSGTR****Register type**

Peg

**Description**

Facility reject messages transmitted

Facility reject messages transmitted (REJMSGTR) counts facility reject messages that do not terminate at the DMS but are transmitted to another switch. Both incoming and outgoing trunk groups are incremented.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJNORTX****Register type**

Peg

**Description**

Facility reject messages originated due to no routing translation

Facility reject messages originated due to no routing translation (REJNORTX) counts facility reject messages that originate at the DMS because no translation was found for the destination address.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**REJRTUNA****Register type**

Peg



**Description**

Facility reject messages originated due to route unavailable

Facility reject messages originated due to route unavailable (REJRTUNA) counts facility reject messages that originate at the DMS due to network failure.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

## PRASERV

### Description

OM group Primary Rate Access Service (PRASERV) provides an overview of the message traffic generated on the PRI D channel of each trunk group by connectionless signaling on PRI. Specifically, it shows the number of times ANI delivery was attempted during an all trunks busy (ATB) condition. It has a single register, ANIATBDA (ANI all trunks busy delivery attempted).

The following table lists the key and info fields associated with OM group PRASERV:

| Key field                                         | Info field |
|---------------------------------------------------|------------|
| COMMON_LANGUAGE_NAM<br>E identifies a trunk CLLI. | None       |

### Related functional groups

There are no functional groups associated with OM group PRASERV.

### Registers

The following table lists the registers associated with OM group PRASERV and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PRASERV

| Register name            | Measures                               |
|--------------------------|----------------------------------------|
| <a href="#">ANIATBDA</a> | ANI all trunks busy delivery attempted |

#### ANIATBDA

##### Register type

Peg

##### Description

Each occurrence of ANI delivery attempted during an ATB condition causes the ANIATBDA peg to increment.

##### Associated registers

None

**Extension registers**

None

**Associated logs**

None

## Presence

### Description

This OM group captures various counters describing the operation of the Presence service.

The following table lists the key and info fields associated with OM group Presence.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

The following functional groups are related to OM group Presence:

- Session Manager

### Registers

The following table lists the registers associated with OM group Presence and what they measure. For a description of a register, click on the register name.

#### Registers for OM group Presence

| Register name                          | Measures                  |
|----------------------------------------|---------------------------|
| <a href="#">barredSubscribe</a>        | barred subscribes         |
| <a href="#">declineSubscribe</a>       | declined subscribes       |
| <a href="#">notifiedPresence</a>       | notified presence         |
| <a href="#">successSubscribe</a>       | successful subscribes     |
| <a href="#">throttleSubscribe</a>      | throttle subscribes       |
| <a href="#">throttleNotifySelfOnly</a> | throttle notify self only |
| <a href="#">throttleNotifyAll</a>      | throttle notify all       |

**barredSubscribe**

**Register type**

Peg

**Description**

Number of unauthorized subscriptions as defined by service provider provisioned barring list.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**declineSubscribe****Register type**

Peg

**Description**

Number of subscriptions banned by the user directly.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**notifiedPresence****Register type**

Peg

**Description**

Number of presence notifications that were sent containing actual presence information (empty notifications are not tracked).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**successSubscribe****Register type**

Peg

**Description**

Number of successful subscriptions.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**throttleSubscribe****Register type**

Peg

**Description**

Number of subscriptions that were not processed due to per user subscription limit per domain.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**throttleNotifySelfOnly****Register type**

Peg

**Description**

This OM is pegged every time the system does not send out notifications to non-self subscriptions because of a presence state change during minor overload. This is pegged once for the entire state change, and does not reflect the actual number of NOTIFY messages that were not sent out.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**throttleNotifyAll****Register type**

Peg

**Description**

This OM is pegged every time the system does not send out any notifications, including self-subscriptions because of a presence state change during major or severe overload. This is pegged once for the entire state change, and does not reflect the actual number of NOTIFY messages that were not sent out.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## Presence\_Event\_Report

### Description

The Presence Event Report OM group tracks the behavior of the various presence events that are processed by the SESM. Each of the rows in the report represents one of the eight presence event types as well as a total row.

Here is a brief description of the various presence events:

- Activity – A client has indicated that it has detected user activity (keyboard/mouse).
- End Call – A client has ended a call (excludes collaboration sessions).
- Inactive – A client has indicated that it has not detected user activity (keyboard/mouse) for an extended period of time.
- Login – A client has logged into the SESM.
- Logout – A client has logged out from the SESM.
- Manual – A client has indicated a presence state change through manual intervention (user interaction with the client interface).
- New Call – A client has entered into a stable call (excludes collaboration sessions).

The following table lists the key and info fields associated with OM group Presence\_Event\_Report.

| Key field | Info field |
|-----------|------------|
| None      | None       |



## Related functional groups

The following functional groups are related to OM group Presence\_Event\_Report:

- Session Manager

## Registers

The following table lists the registers associated with OM group Presence\_Event\_Report and what they measure. For a description of a register, click on the register name.

### Registers for OM group Presence\_Event\_Report

| Register name             | Measures  |
|---------------------------|-----------|
| <a href="#">Created</a>   | created   |
| <a href="#">Processed</a> | processed |
| <a href="#">Optimized</a> | optimized |
| <a href="#">Queued</a>    | queued    |
| <a href="#">Parked</a>    | parked    |

#### Created

##### Register type

Peg

##### Description

The number of events of that type that have been created in the system. This gives the operator an idea of the relative frequency of occurrence for that presence event type.

##### Associated registers

None

##### Extension registers

None

##### Associated logs

None

#### Processed

##### Register type

Peg

**Description**

The number of events of that type that have been processed by the presence event processor. Just because a presence event is created does not mean that it is guaranteed to ever be processed. It may be eliminated from consideration because of an opposing presence event (see optimized register).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**Optimized****Register type**

Peg

**Description**

The number of events of that type that have been optimized by the presence vent processor. An event is optimized when an opposing presence vent is processed that nullifies the presence vent change that would have taken place. For instance, if a new call event is processed and the presence event processor sees that there is an opposing end call event in the queue or parked, then there is no further point in processing either event. The two events cancel each other out.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**Queued****Register type**

Usage

**Scan rate**

**Description**

The number of events that are currently in the presence event processor queue waiting to be processed. This OM represents a snapshot view of the current presence event queue depth.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**Parked****Register type**

Usage

**Scan rate****Description**

The number of events that have been initially processed, but must wait for the presence guard timer to expire before being processed. These events are “parked” waiting for the guard timer to expire. When the guard timer expires, they re-enter the presence event queue (which does not cause a second pegging of the processed counter for that event).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## PRIMWIC

### Description

PRIMWIC is a multiple tuple operational measurement (OM) group that collects and displays counts for every primary rate interface (PRI) access interface that has a Message Waiting Indicator (MWI) Control feature provisioned. The PRIMWIC OM group is maintained on a 30-minute basis.

The following table lists the key and info fields associated with OM group PRIMWIC.

| Key field                                            | Info field |
|------------------------------------------------------|------------|
| Logical terminal identifier (LTID) for PRI interface | none       |

### Related functional groups

NI0-PRI Message Services (Functional Group NI-00037) is associated with OM group PRIMWIC.

### Registers

The following table lists the registers associated with OM group PRIMWIC and what they measure. For a description of a register, click on the register name.

#### Registers for OM group PRIMWIC

| Register name             | Measures                                             |
|---------------------------|------------------------------------------------------|
| <a href="#">ACTATT</a>    | message waiting indication [MWI] activation attempts |
| <a href="#">DEACTATT</a>  | MWI deactivation attempts                            |
| <a href="#">UNSUCTACT</a> | unsuccessful MWI activation attempts                 |
| <a href="#">UNSUCTDAC</a> | unsuccessful MWI deactivation attempts               |
| <a href="#">TASKRFSD</a>  | MWI control task refused                             |
| <a href="#">TMREXPRD</a>  | MWI control timer expired                            |
| <a href="#">NOTFUNAV</a>  | notification unavailable to destination DN           |

**Registers for OM group PRIMWIC**

| Register name           | Measures             |
|-------------------------|----------------------|
| <a href="#">RESUNAV</a> | resource unavailable |
| <a href="#">INVARG</a>  | invalid argument     |

**ACTATT****Register type**

Peg

**Description**

Register ACTATT (message waiting indication [MWI] activation attempts) is the total number of MWI activation attempts received from message storage and retrieval (MSR) using the MWI Control feature.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DEACTATT****Register type**

Peg

**Description**

Register DEACTATT (MWI deactivation attempts) is the total number of MWI deactivation attempts received from an MSR using the MWI Control feature.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**UNSUCTACT****Register type**

Peg

**Description**

Register UNSUCTACT (unsuccessful MWI activation attempts) is the total number of MWI unsuccessful activation attempts received from an MSR using the MWI Control feature.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**UNSUCDAC****Register type**

Peg

**Description**

Register UNSUCDAC (unsuccessful MWI deactivation attempts) is the total number of MWI unsuccessful deactivation attempts received from an MSR using the MWI Control feature.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**TASKRFSD****Register type**

Peg

**Description**

Register TASKRFSD (MWI control task refused) is pegged when the signaling system 7 (SS7) network is either currently overloaded or cannot handle the request.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**TMREXPRD****Register type**

Peg

**Description**

Register TMREXPRD (MWI control timer expired) is pegged when timer MWI-T1 (i.e. OFCENG parameter NMS\_ACKNOWLEDGEMENT\_TIMEOUT) expires before indicating a successful or unsuccessful completion of attempt.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**NOTFUNAV****Register type**

Peg

**Description**

Register NOTFUNAV (notification unavailable to destination DN) is pegged when notification cannot be provided to the destination DN for some short term reason, such as when a line is temporarily out of service.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**RESUNAV****Register type**

Peg

**Description**

Register RESUNAV (resource unavailable) is pegged when the PRI with MWI Control subscription exceeds the number of simultaneously allowable unacknowledged MWI control requests.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**INVARG****Register type**

Peg

**Description**

Register INVARG (invalid argument) is pegged when an argument, such as control type, destination DN, or MSR ID, provided in the MWI control request is invalid.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



## PRISVCS

### Description

PRI services (PRISVCS)

The OM group PRISVCS records information on PRI SERVICES. Two-B-channel transfer (TBCT) associates with the ISDN NI-2-PRI interface. The TBCT uses this OM group. NI-2 PRI interfaces can terminate on the following:

- Intelligent Peripherals (IP)
- Private Branch Exchanges (PBX)
- Customer Premises Equipment (CPE)

TBCT allows a user on a NI-2-PRI interface to request the SSP to connect two calls on the interface that are not related. The SSP releases the B-channels to the NI-2-PRI interface after the SSP directly connects the two users with a speech path. The NI-2 PRI B-channels that the SSP releases are available for other calls.

For each LTID (NI-2-PRI trunk group) supplied in table LTDATA, with the TBCT option, the system creates two OM registers.

The OM group PRISVCS contains two registers that count the following NI-2 PRI services events:

- One register counts the total number of TBCT attempts.
- One register counts the total number of TBCT successes.

The following table lists the key and info fields associated with OM group PRISVCS.

| Key field                     | Info field                                     |
|-------------------------------|------------------------------------------------|
| <COMMON_LANGUAGE_NAME (CLLI)> | Contains three parts: TRKDIR, NCCT, and NWCCT. |

The TRKDIR is the trunk group direction. The fixed TRKDIR for TRK are as follows:

- IC—incoming trunk
- OG—outgoing trunk
- 2W—two-way trunk

The NCCT is the total number of trunk circuits in the group.

The NWCCT is the number of trunk circuits available for service at the end of the reporting period.

## Related functional groups

There are no related functional groups.

## Registers

The following table lists the registers associated with OM group PRISVCS and what they measure. For a description of a register, click on the register name.

### Registers for OM group PRISVCS

| Register name           | Measures                                             |
|-------------------------|------------------------------------------------------|
| <a href="#">TBCTATT</a> | Two-B-Channel Transfer (TBCT) attempts               |
| <a href="#">TBCTSUC</a> | Two-B-Channel Transfer (TBCT) successful connections |

#### TBCTATT

##### Register type

Peg

##### Description

Two-B-Channel Transfer (TBCT) attempts (TBCTATT)

Register TBCTATT stores the number of TBCT attempts the SSP recognizes.

##### Associated registers

[TBCTSUC](#)

##### Extension registers

There are no extension registers.

##### Associated logs

There are no associated logs.

#### TBCTSUC

##### Register type

Peg

**Description**

Two-B-Channel Transfer (TBCT) successful connections (TBCTSUCC)

Register TBCTSUCC stores the total number of successful TBCT connections and next B-channel releases.

**Associated registers**

[TBCTATT](#)

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

---

## PRKOM

---

### Description

Call park operational measurement

PRKOM provides information on feature usage, traffic measurements, and failures due to software and hardware resource provisioning for the integrated business network (IBN) station features Call Park (CPK) and Directed Call Park (DCPK).

Call Park allows a 500/2500 set or a directory number (DN) appearance on a business set to park calls against its own DN. The parked calls can be retrieved from any station by dialing the call park retrieve access code, or by activating the call park key and dialing the DN against which the call is parked.

Registers [CPKSUCC](#), [CPKFELIM](#), [CPKRCLL](#), and [CPKABAN](#) are incremented for each subscriber group. CPKFEXT and CPKFOVF are incremented for the system.

Directed Call Park allows 500/2500 and business sets to park calls against any valid IBN station DN. The assignment of a security code to a DN prevents retrieval of calls that are parked against that DN unless a valid security code is entered. IBNGRP\_SECINVAL counts the number of invalid security codes that are dialed.

On business sets, the recall is always presented to the DN that parked the call. A recall to a busy station causes the recall timer to be reset.

The party attempting to retrieve a parked call must be in the same subscriber group as the party against which the call was originally parked. An invalid attempt to retrieve a parked call is routed to reorder treatment. Invalid attempts include:

- parked party abandoned before retrieval attempt
- entry of invalid security codes
- resources unavailable (network connection, for example)

A parked call may still be retrieved during the recall to the parking party, provided the recall remains unanswered.

Multiple Appearance Directory Number (MADN) group members with either Single Call Arrangement (SCA) or Multiple Call Arrangement (MCA) have access to DCPK. Since all members of a MADN group share the same DN, only one call may be parked for each MADN group

at any time. DCPK recall rings only the station of the MADN member that originally parked the call. For SCA members, the recall occurs only if the group is idle.

The registers for CPK, except CPKSUCC, also monitor the events in DCPK. When a call is successfully parked by DCPK, DCPKSUCC is incremented.

OM group PRKOM provides one tuple for each subscriber group. The following table lists the key and info fields associated with OM group PRKOM.

| Key field | Info field                                                                                              |
|-----------|---------------------------------------------------------------------------------------------------------|
| none      | OMIBNGINFO identifies the name of the subscriber group, as defined in field CUSTNAME of table CUSTHEAD. |

Call Park is implemented through option OPTLIST in table IBNLINES, or when the field FEAT is assigned PRK in table KSETFEAT.

Directed Call Park is implemented through option OPTLIST in table IBNLINES, or when the field FEAT is assigned DCPK in table KSETFEAT.

The number of agents that can use this feature at one time is specified by parameter FTRQAGENTS in table OFCENG.

The number of call process wakeups in the system is specified by parameter NUMCPWAKE in table OFCENG.

## Related functional groups

The following functional groups are associated with OM group PRKOM:

- IBN Integrated Business Network
- 500/2500 Business Set

## Registers

The following table lists the registers associated with OM group PRKOM and what they measure. For a description of a register, click on the register name.

### Registers for OM group PRKOM

| Register name            | Measures                      |
|--------------------------|-------------------------------|
| <a href="#">CPKABAN</a>  | Call park abandon             |
| <a href="#">CPKFLIM</a>  | Call park failure limit       |
| <a href="#">CPKRCLL</a>  | Call park recall              |
| <a href="#">CPKSUCC</a>  | Call park successful          |
| <a href="#">DCPKSUCC</a> | Directed call park successful |

#### CPKABAN

**Register type**

Peg

**Description**

Call park abandon

CPKABAN is incremented when a parked call is abandoned before it is retrieved or before the recall is answered.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

#### CPKFLIM

**Register type**

Peg

**Description**

Call park failure limit

CPKFLIM counts calls that cannot be parked because the maximum number of calls are already parked. Calls that are counted by CPKFLIM are routed to subscriber group overflow treatment.

The maximum number of calls is specified in field CPKMAXNO in table CUSTHEAD.

**Associated registers**

TRMT3\_GCGRO is incremented when a call is routed to the subscriber group overflow treatment.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CPKRCLL****Register type**

Peg

**Description**

Call park recall

CPKRCLL is incremented when a station receives a recall from a parked call.

- The station is recalled for one of two reasons:
- the no-answer recall timer expired before the call was retrieved

the parked party flashed, forcing a recall even though the timer had not expired

The no-answer timer is set in field CPKRECTO in table CUSTSTN.

**Associated registers**

IBNSG\_DARECALL is incremented when an attendant parks and recalls a call.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**CPKSUCC****Register type**

Peg

**Description**

Call park successful

CPKSUCC is incremented when a call is successfully parked in the subscriber group.

The value for this register is zero unless feature package NTX414AA or NTX571AA is present.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.

**DCPKSUCC****Register type**

Peg

**Description**

Directed call park successful

DCPKSUCC counts calls that are successfully parked against any directory number, using the Direct Call Park feature.

This register will read zero unless feature package NTX414AA or NTX517AA are present.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

There are no associated logs.



## PRP

### Description

Preroute peg PRP counts call attempts to specific codes. PRP provides network management statistics that are used for traffic studies. PRP registers indicate when code block (CBK) controls should be implemented.

OM group PRP provides one tuple for each active preroute control. The maximum number of active network management controls is 256.

The following table lists the key and info fields associated with OM group PRP.

| Key field | Info field                                                            |
|-----------|-----------------------------------------------------------------------|
| none      | PRP_OMINFO consists of the following parts: CT, DR, and SNPA, or STS. |

CT is the code type. The fixed CT for PRP are as follows:

- CCODE Country code
- ACODE Area code
- NAC Non-area digits
- PFX Prefix digits

DR is the digit register, the called number code for which the control is effective.

SNPA is the serving number plan area or serving translation scheme that is the origin of calls that are to be monitored by the control. If the code type is CCODE, then SNPA is set to 999.

### Related functional groups

There are no related functional groups.

## Registers

The following table lists the registers associated with OM group PRP and what they measure. For a description of a register, click on the register name.

### Registers for OM group PRP

| Register name          | Measures           |
|------------------------|--------------------|
| <a href="#">PRPCNT</a> | Preroute peg count |

#### PRPCNT

**Register type**

Peg

**Description**

Preroute peg count

PRPCNT counts calls that originated in the SNPA and were directed to the destination code to which PRPCNT applies.

**Associated registers**

There are no associated registers.

**Extension registers**

There are no extension registers.

**Associated logs**

NWM111 is generated when preroute peg controls are activated or deactivated.

## PSN\_ERDC

### Description

OM group Programmable Service Node (PSN) Error — Data Communication Level (PSN\_ERDC) records the number of errors reported by the PSN data communication layer.

The following table lists the key and info fields associated with OM group PSN\_ERDC:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_ERDC.

### Registers

The following table lists the registers associated with OM group PSN\_ERDC and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_ERDC

| Register name            | Measures                                                      |
|--------------------------|---------------------------------------------------------------|
| <a href="#">MSGSIZE</a>  | Service Control Unit (SCU) Message Size Exceeded Maximum Size |
| <a href="#">HBFAIL</a>   | Heartbeat Failure                                             |
| <a href="#">NOSCOMM</a>  | No Service Control Unit (SCU) Communication                   |
| <a href="#">DCOMHDR</a>  | Data Communication Header Error                               |
| <a href="#">UAPPLMBS</a> | User Application Mail Box Problem                             |
| <a href="#">MSGDROP</a>  | Data Communication Drop Internal Message                      |
| <a href="#">EMSGDROP</a> | Data Communication Drop External Message                      |

#### MSGSIZE

Register type

Peg

**Description**

MSGSIZE is pegged when the PSN detects that an SCU message exceeded the maximum size.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN101

**HBFAIL****Register type**

Peg

**Description**

HBFAIL is pegged when the PSN fails to receive a heartbeat from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN100

**NOSCOMM****Register type**

Peg

**Description**

NOSCOMM is pegged when the PSN is unable to establish SCU communication.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN100

**DCOMHDR****Register type**

Peg

**Description**

DCOMHDR is pegged when the PSN rejects a message from the SCU when it discovers an error in the message header.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN101

**UAPPLMBS****Register type**

Peg

**Description**

UAPPLMBS is pegged when the data communication of the PSN encounters a problem sending a message to a user application.

**Associated registers**

None

**Extension registers**

PSN102

**Associated logs**

None

**MSGDROP****Register type**

Peg

**Description**

MSGDROP is pegged when the data communication level of the PSN drops a message sent to it from another PSN source.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN102

**EMSGDROP****Register type**

Peg

**Description**

EMSGDROP is pegged when the data communication of the PSN drops a message sent to it from an outside source.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN101

## PSN\_ERFM

### Description

OM group Programmable Service Node (PSN) Error in Finite State Machine (PSN\_ERFM) records the number of errors reported by the PSN finite state machine.

The following table lists the key and info fields associated with OM group PSN\_ERFM:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_ERFM.

### Registers

The following table lists the registers associated with OM group PSN\_ERFM and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_ERFM (Sheet 1 of 2)

| Register name            | Measures                                    |
|--------------------------|---------------------------------------------|
| <a href="#">SCUTMOUT</a> | SCU Time Out                                |
| <a href="#">UNEXPENF</a> | Unexpected Error Message — Nonfatal         |
| <a href="#">UNEXPEF</a>  | Unexpected Error Message — Fatal            |
| <a href="#">NOUTR</a>    | Universal Tone Receiver (UTR) Unavailable   |
| <a href="#">NOSTR</a>    | Specialized Tone Receiver (STR) Unavailable |
| <a href="#">NOCNF</a>    | Conference Port Unavailable                 |
| <a href="#">UNSUPTRK</a> | Unsupported Trunk Type                      |
| <a href="#">BCINCOMP</a> | Bearer Capability Incompatible              |
| <a href="#">PRMEXTUN</a> | Primitive Extension Block Not Available     |
| <a href="#">SCREXTUN</a> | Scratchpad Extension Block Not Available    |

**Registers for OM group PSN\_ERFM (Sheet 2 of 2)**

| Register name            | Measures                                 |
|--------------------------|------------------------------------------|
| <a href="#">DIGCOLFL</a> | Digit Collection Failed                  |
| <a href="#">AGNAVAIL</a> | Agent Not Available                      |
| <a href="#">PRIMSTFL</a> | Primitive Invalid For Current Port State |
| <a href="#">PROMPTFL</a> | Prompt Failure                           |
| <a href="#">NOIDLMSG</a> | No Idle Message                          |
| <a href="#">MSGFL</a>    | Message Failure                          |
| <a href="#">SFTWERRv</a> | Software Error                           |
| <a href="#">UNSIGTYP</a> | Unsupported Signaling Type               |
| <a href="#">BADAGST</a>  | Bad Agent State                          |
| <a href="#">DUPMSG</a>   | Duplicate Siginfo Message Received       |
| <a href="#">AGNDTKGP</a> | Agent Not Datafilled in Table TRKGRP     |
| <a href="#">MMSIPARM</a> | Missing Mandatory SigInfo Parameter      |

**SCUTMOUT****Register type**

Peg

**Description**

SCUTMOUT is pegged when the SCU new call timer has expired due to no response from the SCU on a new call event.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN201

**UNEXPENF****Register type**

Peg



**Description**

UNEXPENF is pegged when the PSN encounters a non-fatal unexpected peripheral message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN204

**UNEXPEF****Register type**

Peg

**Description**

UNEXPEF is pegged when the PSN receives a fatal unexpected peripheral message.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN204

**NOUTR****Register type**

Peg

**Description**

NOUTR is pegged when the PSN encounters the problem that the UTR is not available for digit collection.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN206

**NOSTR****Register type**

Peg

**Description**

NOSTR is pegged when the PSN encounters the problem that the STR is not available for tone monitoring.

**Associated registers**

None

**Extension registers**

PSN206

**Associated logs**

None

**NOCNF****Register type**

Peg

**Description**

NOCNF is pegged when the PSN finds no conference port available for bridging.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN206

**UNSUPTRK****Register type**

Peg

**Description**

UNSUPTRK is pegged when the destination trunk in a Connect primitive is not a supported type for PSN.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**BCINCOMP****Register type**

Peg

**Description**

BCINCOMP is pegged when PSN tries to connect/reconnect two trunks where the bearer capability of the two trunks is incompatible.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**PRMEXTUN****Register type**

Peg

**Description**

PRMEXTUN is pegged when the PSN finds no primitive extension block available, a software resources problem.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN205

**SCREXTUN****Register type**

Peg

**Description**

SCREXTUN is pegged when the PSN finds no scratchpad extension block available, a software resources problem.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN205

**DIGCOLFL****Register type**

Peg

**Description**

DIGCOLFL is pegged when PSN fails to start/stop digit collection on a port.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**AGNAVAIL****Register type**

Peg

**Description**

AGNAVAIL is pegged when PSN cannot terminate to an agent due to no idle members.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**PRIMSTFL****Register type**

Peg

**Description**

PRIMSTFL is pegged when the PSN receives a primitive that is invalid for the current port state.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN203

**PROMPTFL****Register type**

Peg

**Description**

PROMPTFL is pegged when PSN cannot play a prompt.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**NOIDLMSG****Register type**

Peg

**Description**

NOIDLMSG is pegged when PSN tries to play a message when there are no idle messages available.

**Associated registers**

None

**Extension registers**

None

**Associated logs**  
PSN206

## **MSGFL**

**Register type**  
Peg

**Description**  
MSGFL is pegged when PSN cannot play a message.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
PSN208

## **SFTWERR**

**Register type**  
Peg

**Description**  
SFTWERR is pegged when the PSN encounters an internal error during the processing of a primitive.

**Associated registers**  
None

**Extension registers**  
None

**Associated logs**  
PSN208

## **UNSIGTYP**

**Register type**  
Peg

**Description**  
UNSIGTYP is pegged when the signaling information parameter is sent to a PSN Agent that is not supported for that signaling type.

**Associated registers**  
None

**Extension registers**

None

**Associated logs**

PSN208

**BADAGST****Register type**

Peg

**Description**

BADAGST is pegged when the PSN receives a primitive that is not supported for the current agent state.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**DUPMSG****Register type**

Peg

**Description**

DUPMSG is pegged when a PSN Agent receives multiple IAMs or SETUPs.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**AGNDTKGP****Register type**

Peg

**Description**

AGNDTKGP is pegged when a PSN Agent is not datafilled in table TRKGRP.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208

**MMSIPARM****Register type**

Peg

**Description**

MMSIPARM is pegged when the signaling information parameter that is sent to a PSN Agent is missing.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN208



## PSN\_ERPS

### Description

OM group Programmable Service Node (PSN) Primitive Processing Error (PSN\_ERPS) record the number of errors reported by the PSN primitive processor.

The following table lists the key and info fields associated with OM group PSN\_ERPS:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_ERPS.

### Registers

The following table lists the registers associated with OM group PSN\_ERPS and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_ERPS (Sheet 1 of 2)

| Register name            | Measures                                |
|--------------------------|-----------------------------------------|
| <a href="#">MSMPARM</a>  | Missing Mandatory Parameter             |
| <a href="#">OFRPARM</a>  | Out Of Range Parameter                  |
| <a href="#">PRMUSRMS</a> | Primitive-User Class Mismatch           |
| <a href="#">MBPRDCC</a>  | Mail Box Problem — Data Communication   |
| <a href="#">MBPRPSA</a>  | Mail Box Problem Primitive Server Audit |
| <a href="#">AUDPSF</a>   | Audit Fails To Receive Port Status      |
| <a href="#">AUDPSM</a>   | SCU Port Status Mismatch                |
| <a href="#">PSNRTFL</a>  | Port Not In PSNROUTE                    |
| <a href="#">AGNACT</a>   | Agent Not Active In PSN Call            |
| <a href="#">MAXBREX</a>  | Maximum Ports To Bridge Exceeded        |

**Registers for OM group PSN\_ERPS (Sheet 2 of 2)**

| Register name            | Measures                                      |
|--------------------------|-----------------------------------------------|
| <a href="#">MANDPDEF</a> | Mandatory Parameter Decode Failure            |
| <a href="#">OPPRMDEF</a> | Optional Parameter Decode Failure             |
| <a href="#">NMINNOBP</a> | Not Minimum Number Of Ports To Bridge         |
| <a href="#">PSNMSGFL</a> | Message Index Not in Table PSNMSGIX           |
| <a href="#">PTDNRS</a>   | Port Taken Down Due To Reset Switch Primitive |
| <a href="#">PDRESTWM</a> | Port Down Due To Warm Restart                 |
| <a href="#">DECODEFL</a> | Header Decode Failure                         |
| <a href="#">BADMACRT</a> | Bad Macro Tag                                 |
| <a href="#">MAXPMEXC</a> | Maximum Primitives Exceeded                   |

**MSMPARM****Register type**

Peg

**Description**

MSMPARM is pegged when the PSN detects a Missing Mandatory Parameter error condition from the primitive sent by the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN202

**OFRPARM****Register type**

Peg

**Description**

OFRPARM is pegged when the PSN detects an Out Of Range error condition from the primitive sent by the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN202

**PRMUSRMS****Register type**

Peg

**Description**

PRMUSRMS is pegged when the PSN receives a primitive that does not belong to the user class.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**MBPRDCC****Register type**

Peg

**Description**

MBPRDCC is pegged when the PSN encounters a problem sending an internal message to the data communication mail box.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MBPRPSA****Register type**

Peg

**Description**

MBPRPSA is pegged when the PSN Audit encounters the problem of unavailable mail box resources.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**AUDPSF****Register type**

Peg

**Description**

AUDPSF is pegged when the PSN Audit fails to receive Port Status from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN400

**AUDPSM****Register type**

Peg

**Description**

AUDPSM is pegged when the PSN Audit encounters mismatched Port Status from the SCU and mismatched status of the agent on the PSN.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN400

**PSNRTFL****Register type**

Peg

**Description**

PSNRTFL is pegged when the PSN receives a destination trunk in the Connect primitive that is not found in table PSNROUTE.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**AGNACT****Register type**

Peg

**Description**

AGNACT is pegged when the PSN receives a primitive for an agent that is not active in a PSN call.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**MAXBREX****Register type**

Peg

**Description**

MAXBREX is pegged every time the PSN encounters a bridge primitive with more than the maximum number of ports to bridge.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**MANDPDEF****Register type**

Peg

**Description**

MANDPDEF is pegged every time the PSN encounters mandatory parameter decoding failure.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN202

**OPPRMDEF****Register type**

Peg

**Description**

OPPRMDEF is pegged when the PSN encounters an optional parameter decoding failure.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN202

**NMINNOBP****Register type**

Peg

**Description**

This register is pegged when the PSN encounters too few ports to bridge in the bridge primitive.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**PSNMSGFL****Register type**

Peg

**Description**

PSNMSGFL is pegged when the PSN encounters a primitive with a message index not datafilled in table PSNMSGIX.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN207

**PTDNRS****Register type**

Peg

**Description**

PTDNRS is pegged when a PSN port is taken down due to a reset switch primitive.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN401

**PDRESTWM****Register type**

Peg

**Description**

PDRESTWM is pegged when a port is taken down due to a warm restart.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**DECODEFL****Register type**

Peg

**Description**

DECODEFL is pegged when the PSN detects a header decode failure in a primitive received.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN202

**BADMACRT****Register type**

Peg

**Description**

BADMACRT is pegged when the PSN detects a bad macro tag in a macro received.

**Associated registers**

None



**Extension registers**

None

**Associated logs**

PSN202

**MAXPMEXC****Register type**

Peg

**Description**

MAXPMEXC is pegged when the PSN receives a macro with the number of primitives exceeding the maximum allowed.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN209

## PSN\_FCTR

### Description

OM group PSN Flow Control (PSN\_FCTR) record the number of calls blocked by the PSN under flow control.

The following table lists the key and info fields associated with OM group PSN\_FCTR:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_FCTR.

### Registers

The following table lists the registers associated with OM group PSN\_FCTR and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_FCTR

| Register name            | Measures                               |
|--------------------------|----------------------------------------|
| <a href="#">SCUCDROP</a> | ANI all trunks busy delivery attempted |

#### SCUCDROP

##### Register type

Peg

##### Description

SCUCDROP is pegged every time a call is blocked by the PSN under flow control initiated by SCU.

##### Associated registers

None

**Extension registers**

None

**Associated logs**

None

## PSN\_NOTF

### Description

OM group Programmable Service Node (PSN) Notifications Sent (PSN\_NOTF) are pegged each time the PSN sends an event notification message to the service control unit (SCU). One PSN\_NOTF register exists for each event notification.

The following table lists the key and info fields associated with OM group PSN\_NOTF:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_NOTF.

### Registers

The following table lists the registers associated with OM group PSN\_NOTF and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_NOTF (Sheet 1 of 2)

| Register name            | Measures              |
|--------------------------|-----------------------|
| <a href="#">DIGCOL</a>   | Digits Collected      |
| <a href="#">ERRDET</a>   | Error Detected        |
| <a href="#">INSTCMPL</a> | Instruction Completed |
| <a href="#">MSGPLY</a>   | Message Played        |
| <a href="#">NEWCALL</a>  | New Call              |
| <a href="#">OFFHOOK</a>  | Off Hook              |
| <a href="#">ONHOOK</a>   | On Hook               |
| <a href="#">PORTSTAT</a> | Port Status           |
| <a href="#">RTEUNAV</a>  | Route Not Available   |
| <a href="#">RTESEL</a>   | Route Selected        |

**Registers for OM group PSN\_NOTF (Sheet 2 of 2)**

| Register name            | Measures                |
|--------------------------|-------------------------|
| <a href="#">SIGEVENT</a> | Signaling Event         |
| <a href="#">TONEDET</a>  | Tone Detected           |
| <a href="#">CURTMDY</a>  | Current Time of the Day |
| <a href="#">INSERVCE</a> | In Service              |
| <a href="#">QRYPORT</a>  | Query Port              |

**DIGCOL****Register type**

Peg

**Description**

DIGCOL is pegged every time the PSN sends a Digits Collected event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRDET****Register type**

Peg

**Description**

ERRDET is pegged every time the PSN sends an Error Detected event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**INSTCMPL****Register type**

Peg

**Description**

INSTCMPL is pegged every time the PSN sends an Instruction Completed event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MSGPLY****Register type**

Peg

**Description**

MSGPLY is pegged every time the PSN sends a Message Played event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NEWCALL****Register type**

Peg

**Description**

NEWCALL is pegged every time the PSN sends a New Call event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**OFFHOOK****Register type**

Peg

**Description**

OFFHOOK is pegged every time the PSN sends an Off-Hook event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ONHOOK****Register type**

Peg

**Description**

ONHOOK is pegged every time the PSN sends an On-Hook event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PORTSTAT****Register type**

Peg

**Description**

PORTSTAT is pegged every time the PSN sends a Port Status event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RTEUNAV****Register type**

Peg

**Description**

RTEUNAV is pegged every time the PSN sends a Route Not Available event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RTESEL****Register type**

Peg

**Description**

RTESEL is pegged every time the PSN sends a Route Selected event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None



**SIGEVENT****Register type**

Peg

**Description**

SIGEVENT is pegged every time the PSN sends a Signaling Event event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**TONEDT****Register type**

Peg

**Description**

TONEDT is pegged every time the PSN sends a Tone Detected event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**CURTMDY****Register type**

Peg

**Description**

CURTMDY is pegged every time the PSN sends a Current Time of the Day event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**INSERVCE****Register type**

Peg

**Description**

INSERVCE is pegged every time the PSN sends an In Service event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**QRYPORT****Register type**

Peg

**Description**

QRYPORT is pegged every time the PSN sends a Query Port event notification to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## PSN\_PRIM

### Description

OM group Programmable Service Node (PSN) Primitives Received (PSN\_PRIM) records the number of each service control unit (SCU) primitive received by the PSN.

The following table lists the key and info fields associated with OM group PSN\_PRIM:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_PRIM.

### Registers

The following table lists the registers associated with OM group PSN\_PRIM and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_PRIM (Sheet 1 of 2)

| Register name            | Measures                                |
|--------------------------|-----------------------------------------|
| <a href="#">CDRPT</a>    | Collect Digits and Report               |
| <a href="#">CNECT</a>    | Connect                                 |
| <a href="#">DISCNECT</a> | Disconnect                              |
| <a href="#">HOLD</a>     | Hold                                    |
| <a href="#">MONITOR</a>  | Monitor                                 |
| <a href="#">MUTE</a>     | Mute                                    |
| <a href="#">PLAYMSG</a>  | Play Message                            |
| <a href="#">PPCDRPT</a>  | Play Prompt, Collect Digits, and Report |
| <a href="#">QURYPRT</a>  | Query Port                              |
| <a href="#">RECNECT</a>  | Reconnect                               |

**Registers for OM group PSN\_PRIM (Sheet 2 of 2)**

| Register name            | Measures                    |
|--------------------------|-----------------------------|
| <a href="#">RSETSWCH</a> | Reset Switch                |
| <a href="#">SETBLREC</a> | Set Billing Record          |
| <a href="#">STOPMSG</a>  | Stop Message                |
| <a href="#">XSIGINFO</a> | Transmit Signal Information |
| <a href="#">BRIDGE</a>   | Bridge                      |
| <a href="#">NCALLACC</a> | New Call Accepted           |
| <a href="#">NCALLREJ</a> | New Call Rejected           |
| <a href="#">HEARTBT</a>  | Heartbeat                   |
| <a href="#">ERRDETP</a>  | Error Detected              |
| <a href="#">QURYMDY</a>  | Query Time of Day           |
| <a href="#">SETIPADD</a> | Set IP Address              |
| <a href="#">PRTSTAT</a>  | Port Status                 |
| <a href="#">FLOWCTRL</a> | Flow Control                |

**CDRPT****Register type**

Peg

**Description**

CDRPT is pegged each time the PSN receives a Collect Digits and Report primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**CNECT****Register type**

Peg

**Description**

CNECT is pegged every time the PSN receives a Connect primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**DISCNECT****Register type**

Peg

**Description**

DISCNECT is pegged every time the PSN receives a Disconnect primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**HOLD****Register type**

Peg

**Description**

HOLD is pegged every time the PSN receives a Hold primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MONITOR****Register type**

Peg

**Description**

MONITOR is pegged every time the PSN receives a Monitor primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**MUTE****Register type**

Peg

**Description**

MUTE is pegged every time the PSN receives a Mute primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN400

**PLAYMSG****Register type**

Peg

**Description**

PLAYMSG is pegged every time the PSN receives a Play Message primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PPCDRPT****Register type**

Peg

**Description**

PPCDRPT is pegged every time the PSN receives a Play Prompt, Collect Digits, and Report primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**QURYPORPT****Register type**

Peg

**Description**

QURYPORPT is pegged every time the PSN receives a Query Port primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RECNECT****Register type**

Peg

**Description**

RECNECT is pegged every time the PSN receives a Reconnect primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**RSETSWCH****Register type**

Peg

**Description**

RSETSWCH is pegged every time the PSN receives a Reset Switch primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SETBLREC****Register type**

Peg

**Description**

SETBLREC is pegged every time the PSN receives a Set Billing Record primitive from the SCU.

**Associated registers**

None



**Extension registers**

None

**Associated logs**

None

**STOPMSG****Register type**

Peg

**Description**

STOPMSG is pegged every time the PSN receives a Stop Message primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**XSIGINFO****Register type**

Peg

**Description**

XSIGINFO is pegged every time the PSN receives a Transmit Siginfo primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**BRIDGE****Register type**

Peg

**Description**

BRIDGE is pegged every time the PSN receives a Bridge primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCALLACC****Register type**

Peg

**Description**

NCALLACC is pegged every time the PSN receives a New Call Accepted primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**NCALLREJ****Register type**

Peg

**Description**

NCALLREJ is pegged every time the PSN receives a New Call Rejected primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**HEARTBT****Register type**

Peg

**Description**

HEARTBT is pegged every time the PSN receives a Heartbeat primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**ERRDETP****Register type**

Peg

**Description**

ERRDETP is pegged every time the PSN receives an Error Detected primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

PSN200

**QURYMDY****Register type**

Peg

**Description**

QURYMDY is pegged every time the PSN receives a Query Time of the Day primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SETIPADD****Register type**

Peg

**Description**

SETIPADD is pegged every time the PSN receives a Reset Switch primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**PRTSTAT****Register type**

Peg

**Description**

PRTSTAT is pegged every time the PSN receives a Port Status primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**FLOWCTRL****Register type**

Peg

**Description**

FLOWCTRL is pegged every time the PSN receives a Flow Control primitive from the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## PSN\_USAG

### Description

OM group Programmable Service Node (PSN) Usage (PSNUSAG) records the number of service control unit (SCU) messages received by the PSN, the number of messages the PSN sends to the SCU, and the number of SCU macros received by the PSN.

The following table lists the key and info fields associated with OM group PSN\_USAG:

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

There are no functional groups associated with OM group PSN\_USAG.

### Registers

The following table lists the registers associated with OM group PSN\_USAG and what they measure. For a detailed description of a register, click on the register name.

#### Registers for OM group PSN\_USAG

| Register name            | Measures                                        |
|--------------------------|-------------------------------------------------|
| <a href="#">SMSGRCVD</a> | Service Control Unit Primitive Message Received |
| <a href="#">SMSGSENT</a> | Service Control Unit Event Notification Sent    |
| <a href="#">SMACRCVD</a> | Service Control Unit Macro Received             |

#### SMSGRCVD

##### Register type

Peg

##### Description

SMSGRCVD is pegged for every message the PSN receives from the SCU.

##### Associated registers

None

**Extension registers**

None

**Associated logs**

None

**SMSGSENT****Register type**

Peg

**Description**

SMSGSENT is pegged when the PSN sends a message to the SCU.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**SMACRCVD****Register type**

Peg

**Description**

SMACRCVD is pegged for every macro the PSN receives from the SCU. A macro is a set of primitives.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

## Publish

### Description

This OM group provides counters for the usage of the PUBLISH service.

The following table lists the key and info fields associated with OM group Publish.

| Key field | Info field |
|-----------|------------|
| None      | None       |

### Related functional groups

The following functional groups are related to OM group Publish:

- Session Manager

### Registers

The following table lists the registers associated with OM group Publish and what they measure. For a description of a register, click on the register name.

#### Registers for OM group Publish

| Register name                     | Measures                    |
|-----------------------------------|-----------------------------|
| <a href="#">pub_refresh_succ</a>  | PUBLISH refresh successful  |
| <a href="#">pub_modify_succ</a>   | PUBLISH modify successful   |
| <a href="#">pub_remove_succ</a>   | PUBLISH remove successful   |
| <a href="#">pub_proxy_reject</a>  | PUBLISH proxy rejected      |
| <a href="#">pub_etag_reject</a>   | PUBLISH entity-tag rejected |
| <a href="#">pub_expire_reject</a> | PUBLISH expiration rejected |
| <a href="#">pub_atom_reject</a>   | PUBLISH atomicity rejected  |
| <a href="#">pub_proxied</a>       | PUBLISH proxied             |
| <a href="#">pub_creation_succ</a> | PUBLISH creation successful |



**pub\_refresh\_succ****Register type**

Peg

**Description**

Successful refresh of event state. This means that a PUBLISH was received that caused existing state information to be refreshed (much like a contact registration).

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_modify\_succ****Register type**

Peg

**Description**

Successful modification of event state. This means that an existing piece of event state information was successfully changed to a new value. This helps to verify that EPAs are modifying existing event state information by including the correct etag information rather than creating new event state information.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_remove\_succ****Register type**

Peg

**Description**

Successful removal of event state. This means that an existing piece of event state information was successfully removed from the system as opposed to it simply expiring.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_proxy\_reject****Register type**

Peg

**Description**

Rejection of PUBLISH requests due to the resource not residing in a local domain. This OM can be used to determine if non-subscribers are attempting to address the ESC with a resource that cannot be proxied and is otherwise unknown to the system.

**Associated registers**[pub\\_proxied](#)**Extension registers**

None

**Associated logs**

None

**pub\_etag\_reject****Register type**

Peg

**Description**

Rejection of PUBLISH requests due to entity-tag checking (400, 412 responses). This OM can be used to determine if EPAs are correctly tracking the entity-tags present in the responses to their PUBLISH requests for protocol verification.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_expire\_reject****Register type**

Peg

**Description**

Rejection of PUBLISH requests due to invalid expiration interval (423 responses). This OM shows how many publications were rejected due to expiration intervals that were considered below the minimum expiration threshold defined by the operator.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_atom\_reject****Register type**

Peg

**Description**

Rejection of PUBLISH requests due to back-end (atomicity) processing failures (504 responses). This OM shows how many publications were rejected due to atomicity errors on the Session Manager where the publication could only be partially processed.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None

**pub\_proxied****Register type**

Peg

**Description**

Proxying of PUBLISH requests to another ESC for a local domain. This OM can be used to determine if the network topology is laid out

correctly or that publications are not being proxied excessively/unexpectedly through a particular Session Manager.

**Associated registers**

[pub\\_proxy\\_reject](#)

**Extension registers**

None

**Associated logs**

None

**pub\_creation\_succ****Register type**

Peg

**Description**

Successful creation of event state. This means that a PUBLISH was received and caused new new state information for a particular resource to be created. This OM may be used to verify that EPAs are not creating new event state information excessively.

**Associated registers**

None

**Extension registers**

None

**Associated logs**

None