CHAPTER 11

Diagnostic Group

Diagnostics refers to the group of functions used to test or view the operating status of various system components. The following section lists the card diagnostic group objects that run diagnostic tests on system cards. For further information, refer to the documentation for the specific card.

In case of a failure, contact your Cisco Systems TAC.

Warning **All tests are for diagnostic purposes ONLY and are invasive. DO NOT perform any tests without direction from your Cisco Systems TAC.**

Card Diagnostic Group

The Card Diagnostic Group enables the user to configure the diagnostic tests on the system cards.

diagSessionTable

{diagnostics 1}

Description

The list of the diagnostic sessions.

Object Identifier

1.3.6.1.4.1.886.1.10.1

Data Type

Sequence of DiagSessionEntry

Access Policy

Not accessible

Status

Mandatory

diagSessionEntry

{diagSessionTable 1}

Description

A diagnostic session.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1

Data Type

DiagSessionEntry

Access Policy

Not accessible

Status

Mandatory

Index

diagSessionIndex

DiagSessionEntry

Sequence

diagSessionIndex	CardIndex
diagTestFunction	Integer
diagTestSubfunction	Integer
diagTestSubsubfunction	Integer
diagTestProcess	Integer
diagTestArgNum	Integer
diagTestResultCode	Integer
diagTestResultNum	Integer
diagSessionState	Integer
diagSessionErrorStatus	Integer
diagSessionOwnerString	OwnerString
diagSessionEntry Status	Entry Status

diagSessionIndex

{diagSessionEntry 1}

Description

Index for this diagnostic session. The index is the same as the index for the card on which the diagnostic session is performed.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.1

Data Type

CardIndex

Access Policy

Read only

Status

Mandatory

diagTestFunction

{diagSessionEntry 2}

Description

The diagnostic test function to perform.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.2

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read-write

Status

Mandatory

diagTestSubfunction

{diagSessionEntry 3}

Description

The diagnostic test subfunction to perform.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.3

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read-write

Status

Mandatory

diagTestSubsubfunction

{diagSessionEntry 4}

Description

The diagnostic test subsubfunction to perform.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.4

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read-write

Status

Mandatory

diagTestProcessor

{diagSessionEntry 5}

Description

The processor on which to perform the diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.5

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read-write

Status

Mandatory

diagTestArgNum

{diagSessionEntry 6}

Description

The number of arguments in diagTestArgTable for this diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.6

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read only

Status

Mandatory

diagTestResultCode

{diagSessionEntry 7}

Description

The result code of a completed diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.7

Data Type

Integer. Possible values are from 0 to 65535. A 0 (zero) value in the result code means the test passed.

Access Policy

Read only

Status

Mandatory

diagTestResultNum

{diagSessionEntry 8}

Description

The number of result values in diagTestResultTable for this diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.8

Data Type

Integer. Possible values are from 0 to 65535.

Access Policy

Read only

Status

Mandatory

diagSessionState

{diagSessionEntry 9}

Description

The state that the session is in. This value can only be set to “starting” if the session is idle, or “aborting” if the session is running.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.9

Data Type

Integer. Possible values and their meaning are shown in the following list:

Value	Setting	Conditions
1	inactive	None
2	activating	None
3	idle	None
4	starting	Must be in idle state
5	running	None
6	aborting	Must be running state

Access Policy

Read-write

Status

Mandatory

diagSessionErrorStatus

{diagSessionEntry 15}

Description

Registers the last error that occurred on this diagSession entry.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.15

Data Type

Integer. The possible values and their meaning are shown in the following list:

Value	Meaning
1792	diagnostic-session-already-exists
1793	diagnostic-session-is-not-idle
1794	illegal-value
1795	illegal-argument-index
1796	no-card-exists
1797	illegal-state-for-state-change
1798	test-denied
1799	already-running-a-test
1800	diagnostic-state-corrupted
1801	test-aborted-by-user

Value	Meaning
1802	internal-allocation-error
1803	internal-queue-error
1804	internal-mailbox-error
1805	timeout-waiting-for-card-response

Access Policy

Read only

Status

Mandatory

diagSessionOwnerString

{diagSessionEntry 16}

Description

The entity that configured this object and is therefore using the assigned object.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.16

Data Type

OwnerString

Access Policy

Read-write

Status

Mandatory

diagSessionEntryStatus

{diagSessionEntry 17}

Description

The status of this entry. This object also determines whether the corresponding entries in the diagtestArgTable (see the “Diagnostic Test Argument Table” section on page 11-8) can be modified.

Object Identifier

1.3.6.1.4.1.886.1.10.1.1.17

Data Type

Entry Status

Access Policy

Read-write

Status

Mandatory

Diagnostic Test Argument Table

The diagnostic test argument table contains objects that enable you to establish the arguments by which system cards are diagnosed.

diagTestArgTable

{diagnostics 2}

Description

The table of arguments for diagnostics tests. Each diagnostic test in a session is indexed in the diagSessionTable with the diagSessionIndex object. There is an entry in this table that lists the modified arguments for each diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.2

Data Type

Sequence of DiagTestArgEntry

Access Policy

Not accessible

Status

Mandatory

diagTestArgEntry

{diagTestArgTable 1}

Description

A set of arguments for a test. You can modify the arguments for a test only when the diagSessionEntryStatus object is in the underModification state.

Object Identifier

1.3.6.1.4.1.886.1.10.2.1

Data Type

DiagTestArgEntry

Access Policy

Not accessible

Status

Mandatory

Index

{diagSessionIndex, diagTestArgIndex}

DiagTestArgEntry

Sequence

diagTestArgIndex	Integer
diagTestArgValue	Integer

diagTestArgIndex

{diagTestArgEntry 1}

Description

The index for a specific argument in a diagnostic test. This object, together with the diagSessionIndex, identifies an argument.

Object Identifier

1.3.6.1.4.1.886.1.10.2.1.1

Data Type

Integer

Access Policy

Read only

Status

Mandatory

diagTestArgValue

{diagTestArgEntry 2}

Description

The value for a specific argument in a diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.2.1.2

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read-write

Status

Mandatory

Diagnostic Test Result Table

The objects in the diagnostic test result table enable you to access the test results from a completed diagnostic test. These test results are valid only when the test fails.

diagTestResultTable

{diagnostics 3}

Description

The table of results from diagnostic tests. For each diagnostic test in a session there is an entry in the table listing the returned results of the completed test. (The diagSessionTable is indexed by the diagSessionIndex.)

Object Identifier

1.3.6.1.4.1.886.1.10.3

Data Type

Sequence of DiagTestResultEntry

Access Policy

Not accessible

Status

Mandatory

diagTestResultEntry

{diagTestResultTable 1}

Description

A set of results from a test.

Object Identifier

1.3.6.1.4.1.886.1.10.3.1

Data Type

DiagTestResultEntry

Access Policy

Not accessible

Status

Mandatory

Index

{diagSessionIndex, diagTestResultIndex}

DiagTestResultEntry

Sequence

diagTestResultIndex	Integer
diagTestResultValue	Integer

diagTestResultIndex

{diagTestResultEntry 1}

Description

Identifies the location of the diagTestResultEntry in the diagTestResultTable from a completed diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.3.1.1

Data Type

Integer

Access Policy

Read only

Status

Mandatory

diagTestResultValue

{diagTestResultEntry 2}

Description

The value for a specific result from a completed diagnostic test.

Object Identifier

1.3.6.1.4.1.886.1.10.3.1.2

Data Type

Integer. Possible values are from 0 to 255.

Access Policy

Read only

Status

Mandatory

diagSessionTableLastModified

{diagnostic 4}

Description

The time, in hundredths of a second, since the diagnostic session table was last modified. You can poll this object to trigger rediscovery.

Object Identifier

1.3.6.1.4.1.886.1.10.4

Data Type

TimeTicks

Access Policy

Read only

Status

Mandatory

Set Up Path Function

The Set Up Path function provides the ability to create a voice path between two ports (referred to as A and B). A voice path is a physical, system-switched connection that allows the transfer of in-band signaling and/or voice energy. In-band signaling can consist of MF or DTMF digits, tones, or voice prompts. Ports can be specified by hardware or software address. Port A and Port B do not have to be linked into the same resource chain in order to create a voice path.

System resources are divided into two categories: senders, which are the source of voice and in-band signaling information, and receivers, which listen to that information. When you create a voice path, you must designate one port as the sender and one port as the receiver.

Valid senders can include tone channels, DVC ports, IPRC ports, SLIC ports, DID ports, UTC ports, single span T1 channels, single span E1 channels, PRI B-channels, E+M ports, DCC ports, four span T1 channels, four span E1 channels, and MVDC T1 channels.

Valid receivers can include DRC ports, MRC ports, CPA ports, MFCR2 ports, SLIC ports, DID ports, UTC ports, single span T1 channels, single span E1 channels, PRI B-channels, E+M ports, DCC ports, four span T1 channels, four span E1 channels, and MVDC T1 channels.

If both ports specified can send and receive, a two-way voice path is created. Otherwise, the voice path is one way. Up to eight voice paths can exist simultaneously.

For further information pertaining to the Set Up Path function, refer to the *Cisco VCO/4K System Administrator's Guide.*

You can also create paths using the host API Voice Path Control ($66) command. For more information, refer to the *Cisco VCO/4K Extended Programming Reference* or *Cisco VCO/4K Standard Programming Reference.*

The following MIB objects are available to control Set Up Path functions through SNMP.

setUpPathTable

{diagnostics 5}

Description

The list of paths.

Object Identifier

1.3.6.1.4.1.886.1.10.5

Data Type

Sequence of SetUpPathEntry

Access Policy

Not accessible

Status

Mandatory

setUpPathEntry

{setUpPathTable 1}

Description

A path entry. Up to a maximum of 8 path entries may be defined. Paths can only be added or deleted.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1

Data Type

SetUpPathEntry

Access Policy

Not accessible

Status

Mandatory

setUpPathIndex

{setUpPathEntry 1}

Description

An index into the setUpPath table. Paths must be added to this table sequentially.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.1

Data Type

Integer

Access Policy

Read only

Status

Mandatory

setUpPathPortAddressA

{setUpPathEntry 2}

Description

Specifies the software address (hexadecimal logical identifier) of the first port in the path.The port can also be specified by using the Rack, Level, Slot, and Port objects. If specified, this object takes precedence over the Rack, Level, Slot, and Port objects.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.2

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathRackA

{setUpPathEntry 3}

Description

The rack (R) that the card of the first port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.3

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathLevelA

{setUpPathEntry 4}

Description

The level (L) that the card of the first port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.4

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathSlotA

{setUpPathEntry 5}

Description

The slot (S) that the card of the first port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.5

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathSpanA

{setUpPathEntry 6}

Description

The span that the card of the first port occupies. For single span cards, specify 0.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.6

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathPortA

{setUpPathEntry 7}

Description

The port (P) in the card that is involved in the path.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.7

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathPortAddressB

{setUpPathEntry 8}

Description

Specifies the software address (hexadecimal logical identifier) of the other port in the path. The port can also be specified by using the Rack, Level, Slot, and Port objects. If specified, this object takes precedence over the Rack, Level, Slot, and Port objects.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.8

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathRackB

{setUpPathEntry 9}

Description

The rack (R) that the card of the other port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.9

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathLevelB

{setUpPathEntry 10}

Description

The level (L) that the card of the other port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.10

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathSlotB

{setUpPathEntry 11}

Description

The slot (S) that the card of the other port occupies.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.11

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathSpanB

{setUpPathEntry 12}

Description

The span that the card of the other port occupies. For single span cards, specify 0.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.12

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathPortB

{setUpPathEntry 13}

Description

The port (P) in the card that is involved in the path.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.13

Data Type

Integer

Access Policy

Read-write

Status

Mandatory

setUpPathErrorStatus

{setUpPathEntry 14}

Description

Registers the last error that occurred on this entry. The errors and their meanings are:

String	Meaning
paNotAssigned	Port Address is not assigned
twoRcvError	Can't connect 2 'Rcv Only' Ports
twoXmitError	Can't connect 2 'Xmit Only' Ports
portBadState	Port is not in correct state
cardOos	Card containing Port Address is OOS
noPathOnSbyDtg	Cannot set a path on standby DTG
pathComplete	(not used)
pathDeleted	(not used)
priDChanSel	Invalid PRI/E1 Channel selected
cannotRunOnSby	Cannot run on standby
maxPathsHit	Maximum path limit exceeded
undefinedPath	Undefined path
invalidKey	(not used)
invalidOper	(not used)
paOutOfRange	Port Address is out of range
cantBeZero	Port must be greater than zero
emptyRLSP	Empty R L S P
invalidRLSP	Invalid R L S P
twoManyPorts	Not that many ports on board
noToneCard	No tone card in system
invalidPa	Invalid Port Address
missingInfo	Insufficient information to complete request
invalidPathId	Indices may have values 1 -8 only
pathAlreadyExists	Path already exists for this index
pathDoesNotExist	Path does not exist for this index

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.14

Data Type

Integer. Valid values are:

Value	Error
8448	paNotAssigned
8449	twoRcvError
8450	twoXmitError
8451	portBadState
8452	cardOos
8453	noPathOnSbyDtg
8454	pathComplete
8455	pathDeleted
8456	priDChanSel
8457	cannotRunOnSby
8458	maxPathsHit
8459	undefinedPath
8460	invalidKey
8461	invalidOper
8462	paOutOfRange
8463	cantBeZero
8464	emptyRLSP
8465	invalidRLSP
8466	twoManyPorts
8467	noToneCard
8468	invalidPa
8469	missingInfo
8470	invalidPathId
8471	pathAlreadyExists
8473	unknownErr
8472	pathDoesNotExist

Access Type

Read-only

Status

Mandatory

setUpPathOwnerString

{setUpPathEntry 13}

Description

The entity that configured this entry.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.13

Data Type

OwnerString

Access Policy

Read-write

Status

Mandatory

setUpPathEntryStatus

{setUpPathEntry 14}

Description

The status of this entry.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.14

Data Type

EntryStatus

Access Policy

Read-write

Status

Mandatory

setUpPathTableLastModified

{setUpPathEntry 15}

Description

The time (in hundredths of a second) since the epoch that the setUpPath table was last modified.

Object Identifier

1.3.6.1.4.1.886.1.10.5.1.15

Data Type

TimeTicks

Access Policy

Read-only

Status

Mandatory