AX2500S Software Manual

MIB Reference

For Version 3.5

AX25S-S006X-70

Alaxala

Relevant products

This manual applies to the models in the AX2500S series of switches. It also describes the functionality of version 3.5 of the software for the AX2500S series of switches. The described functionality is that supported by the OS-L2B-A/OS-L2B and the advanced software upgrade license (the "License").

Export restrictions

In the event that any or all ALAXALA products (including technologies, programs and services) described or contained herein are controlled under any of applicable export control laws and regulations (including the Foreign Exchange and Foreign Trade Law of Japan and United States export control laws and regulations), such products shall not be exported without obtaining the required export licenses from the authorities concerned in accordance with the above laws.

Trademarks

Ethernet is a registered trademark of Xerox Corporation.
Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and other countries.
Windows is a registered trademark of Microsoft Corporation in the United States and other countries.
RSA and RSA SecurID are trademarks or registered trademarks of RSA Security Inc. in the United States and other countries.
sFlow is a registered trademark of InMon Corporation in the United States and other countries.
Wake-on-LAN is a registered trademark of IBM Corporation.
MagicPacket is a registered trademark of Advanced Micro Devices,Inc.
Other company and product names in this document are trademarks or registered trademarks of their respective owners.

Reading and storing this manual

Before you use the equipment, carefully read the manual and make sure that you understand all safety precautions. After reading the manual, keep it in a convenient place for easy reference.

Notes

Information in this document is subject to change without notice.

Editions history

January 2013 (Edition 8), AX25S-S006X-70

Copyright

History of Amendments

[Ver.3.5 (Edition 8)]

Summary of amendments

Location and title	Changes
1.1 MIB system diagram	● ax2530sSml group [OS-L2A] was added.
1.2 MIB list	● ax2530sSml group [OS-L2A] was added.
3.4.2 axsL2ldPortTable group	● Descriptions about peer links were added to the axsL2ldPortTable group.
3.11 ax2530sSwitch group (system device model information MIB)	● AX2530S-24TD, AX2530S-48TD, and AX2530S-24S4XD were added to the ax2530sModelType implementation specifiations.
3.12.1 ax2530sChassis group implementation specifications (chassis information)	● AX2530S-24TD, AX2530S-48TD, and AX2530S-24S4XD were added to the ax2530sChassisType implementation specifications. ● AX2530S-24TD, AX2530S-48TD, and AX2530S-24S4XD were added to the ax2530sFanNumber implementation specifications.
3.12.4 ax2530sChassis group implementation specifications (fan information)	● AX2530S-24TD, AX2530S-48TD, and AX2530S-24S4XD were added to the ax2530sFanIndex implementation specifications.
3.12.6 ax2530sPhysLine group implementation specifications ((physical) line information)	● type10GBASE-CU30CM was added to the ax2530sPhysLineConnectorType implementation specifications.
3.14 ax2530sSml group (SML information) [OS-L2A]	● This section was added.
4. Supported MIB Traps	● The following traps related to SML were added: ■ ax2530sSmlStatusFull [OS-L2A] ■ ax2530sSmlStatusStandalone [OS-L2A] ■ ax2530sSmlStatusConflict [OS-L2A] ■ ax2530sSmlPeerlinkNormal [OS-L2A] ■ ax2530sSmlPeerlinkFailure [OS-L2A] ■ ax2530sSmlPeerlinkRecovery [OS-L2A] ■ ax2530sSmlPeerlinkDisconnect [OS-L2A]

In addition to the above changes, minor editorial corrections were made.

[Ver.3.4 (Edition 7)]

Summary of amendments

Location and title	Changes
MIB system diagram	● The sFlow group was added.
MIB list	● The sFlow group was added.

Location and title	Changes
interfaces group	● ifOutUcastPkts for the interfaces group is now supported.
ifMIB	● ifHCOutUcastPkts for ifMIB is now supported.
axsIfStats group	● axsIfStatsOutUcastMegaPkts in the axsIfStats group is now supported.
sFlow group (InMon private MIB)	● This section was added.

In addition to the above changes, minor editorial corrections were made.

[Ver.3.3 (Edition 6)]

Summary of amendments

Location and title	Changes
MIB list	● IPv6 information was added.
ip group (MIB-II)	● IPv6 information was added to the ip group implementation specifications.
icmp group (MIB-II)	● icmp group implementation specifications (#27 to #39) were added.
tcp group (MIB-II)	● tcp group implementation specifications (#22 to #39) were added.
udp group (MIB-II)	● udp group implementation specifications (#9 to #20) were added.
axsAxrpMIB group (Ring Protocol information)	● Descriptions related to master nodes, shared nodes, and multi-fault monitoring functionality were added.

In addition to the above changes, minor editorial corrections were made.

[Ver.3.2 (Edition 5)]

Summary of amendments

Location and title	Changes
MIB list	● The axsVlanTagTranslation group is now supported.
axsVlanTagTranslation group (Tag translation information MIB)	● This section was added.
ax2530sPhysLine group implementation specifications ((physical) line information)	● type10GBASE-ER was added to the ax2530sPhysLineConnectorType implementation specifications.

In addition to the above changes, minor editorial corrections were made.

[Ver.3.2 (Edition 4)]

Summary of amendments

Location and title	Changes
ax2530sSwitch group (system device model information MIB)	● AX2530S-24T4X and AX2530S-48T2X were added to the ax2530sModelType implementation specifications.
ax2530sChassis group implementation specifications (chassis information)	● AX2530S-24T4X and AX2530S-48T2X were added to the ax2530sChassisType implementation specifications. ● AX2530S-24T4X and AX2530S-48T2X were added to the ax2530sFanNumber implementation specifications.
ax2530sChassis group implementation specifications (fan information)	● AX2530S-24T4X and AX2530S-48T2X were added to the ax2530sFanIndex implementation specifications.

In addition to the above changes, minor editorial corrections were made.

[Ver.3.1 (Edition 3)]

Summary of amendments

Location and title	Changes
MIB system diagram	● snmpModules (3) was added.
MIB list	● snmpModules group was added.
snmpModules group	● snmpModules group was added.
ax2530sPhysLine group implementation specifications ((physical) line information)	● type1000BASE-UTP was added to the ax2530sPhysLineConnectorType implementation specifications.

In addition to the above changes, minor editorial corrections were made.

[Ver.3.1 (Edition 2)]

Summary of amendments

Location and title	Changes
interfaces group	● 100BASE-FX and 10GBASE-R were added.
ifMIB	● 100BASE-FX and 10GBASE-R were added.
ax2530sSwitch group (system's model information MIBs)	● AX2530S-24S4X was added to the ax2530sModelType implementation specifications.
ax2530sChassis group implementation specifications (chassis information)	● AX2530S-24S4X was added to the ax2530sChassisType implementation specifications. ● AX2530S-24S4X was added to the ax2530sFanNumber implementation specifications. ● AX2530S-24S4X was added to the ax2530sCriticalAccumRunTime implementation specifications.
ax2530sChassis group implementation specifications (fan information)	● AX2530S-24S4X was added to the ax2530sFanIndex implementation specifications.
ax2530sPhysLine group implementation specifications ((physical) line information)	● type100BASE-FX, type10GBASE-SR, type10GBASE-LR, type10GBASE-CU1M, type10GBASE-CU3M, and type10GBASE-CU5M were added to the ax2530sPhysLineConnectorType implementation specifications. ● SFP+ was added to the ax2530sPhysLineTransceiverStatus implementation specifications.

In addition to the above changes, minor editorial corrections were made.

Preface

Applicable products and software versions

This manual applies to the models in the AX2500S series of switches. It also describes the functionality of version 3.5 of the software for the AX2500S series of switches. The described functionality is that supported by the OS-L2B-A/OS-L2B and the advanced software upgrade license (the "License").

Before you operate the equipment, carefully read the manual and make sure that you understand all instructions and cautionary notes. After reading the manual, keep it in a convenient place for easy reference.

Unless otherwise noted, this manual describes the functionality applicable commonly to AX2500S series switches. The functionalities specific to each model are indicated as follows:

[24T]:

The description applies to the AX2530S-24T switch.

[24T4X]:

The description applies to the AX2530S-24T4X switch.

[48T]:

The description applies to the AX2530S-48T switch.

[48T2X]:

The description applies to the AX2530S-48T2X switch.

[24S4X]:

The description applies to the AX2530S-24S4X switch.

[24TD]:

The description applies to the AX2530S-24TD switch.

[48TD]:

The description applies to the AX2530S-48TD switch.

[24S4XD]:

The description applies to the AX2530S-24S4XD switch.

[10G model]:

The description applies to the AX2530S-24T4X, the AX2530S-48T2X, AX2530S-24S4X, and AX2530S-24S4XD switches.

Unless otherwise noted, this manual describes the functionality for OS-L2B-A/OS-L2B. Functionality related to the Software License Agreement and License Sheet is indicated as follows:

[OS-L2A]:

The description indicates functionality supported by the Software License Agreement and License Sheet.

Corrections to the manual

Corrections to this manual might be contained in the *Release Notes* and *Manual Corrections* that come with the software.

Intended readers

This manual is intended for system administrators who wish to configure and operate a network system that uses the Switch.

Readers must have an understanding of the following:

- The basics of network system management

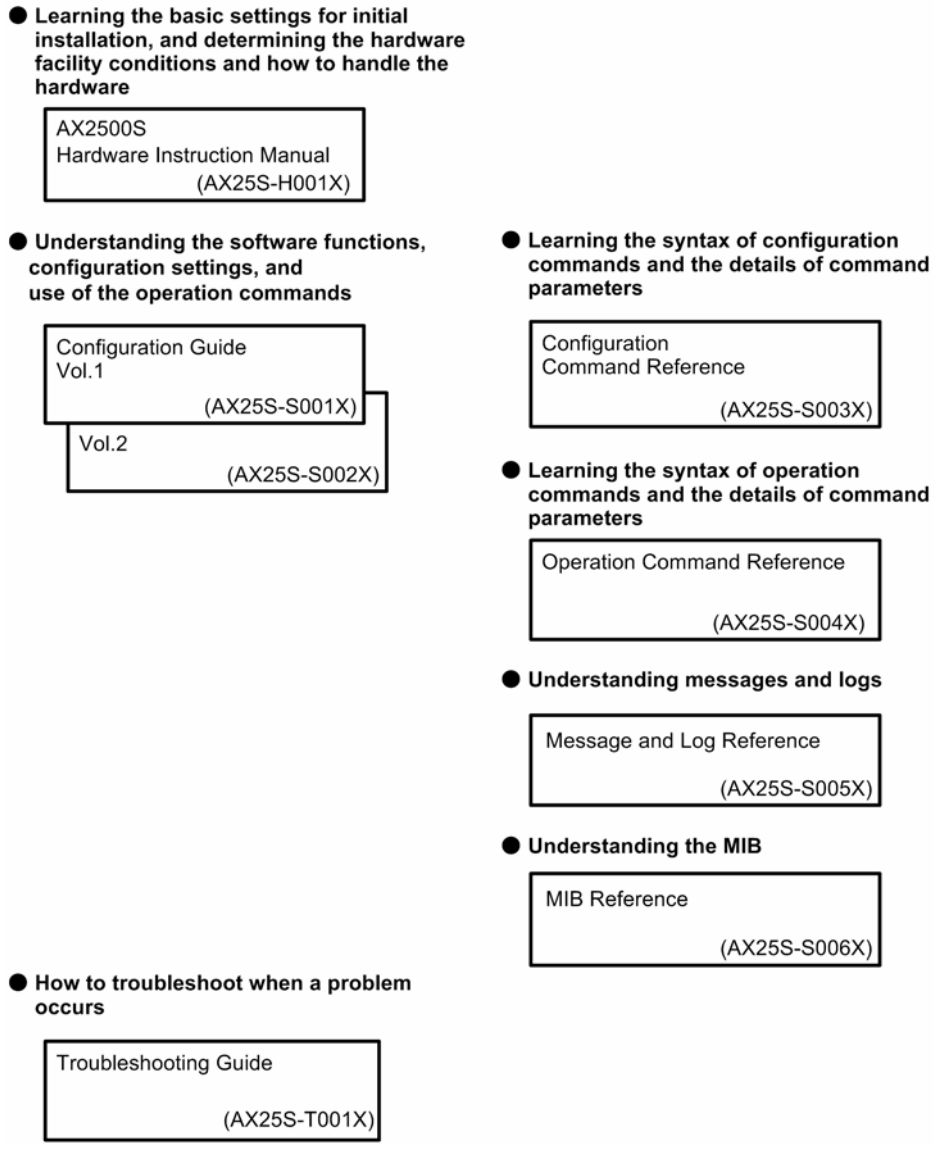
Manual URL

You can view this manual on our website at:

http://www.alaxala.com/en/

Reading sequence of the manuals

The following shows the manuals you need to consult according to your requirements determined from the following workflow for installing, setting up, and starting regular operation of the Switch.



Abbreviations used in the manual

AC	Alternating Current
ACK	ACKnowledge
ADSL	Asymmetric Digital Subscriber Line
ALG	Application Level Gateway
ANSI	American National Standards Institute
ARP	Address Resolution Protocol
AS	Autonomous System
AUX	Auxiliary
BGP	Border Gateway Protocol
BGP4	Border Gateway Protocol - version 4
BGP4+	Multiprotocol Extensions for Border Gateway Protocol - version 4
bit/s	bits per second (can also appear as bps)
BPDU	Bridge Protocol Data Unit
BRI	Basic Rate Interface
CC	Continuity Check
CDP	Cisco Discovery Protocol
CFM	Connectivity Fault Management
CIDR	Classless Inter-Domain Routing
CIR	Committed Information Rate
CIST	Common and Internal Spanning Tree
CLNP	ConnectionLess Network Protocol
CLNS	ConnectionLess Network System
CONS	Connection Oriented Network System
CRC	Cyclic Redundancy Check
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CSNP	Complete Sequence Numbers PDU
CST	Common Spanning Tree
DA	Destination Address
DC	Direct Current
DCE	Data Circuit terminating Equipment
DHCP	Dynamic Host Configuration Protocol
DIS	Draft International Standard/Designated Intermediate System
DNS	Domain Name System
DR	Designated Router
DSAP	Destination Service Access Point
DSCP	Differentiated Services Code Point
DTE	Data Terminal Equipment
DVMRP	Distance Vector Multicast Routing Protocol
E-Mail	Electronic Mail
EAP	Extensible Authentication Protocol
EAPOL	EAP Over LAN
EFM	Ethernet in the First Mile
ES	End System
FAN	Fan Unit
FCS	Frame Check Sequence
FDB	Filtering DataBase
FQDN	Fully Qualified Domain Name

FTTH	Fiber To The Home
GBIC	GigaBit Interface Converter
GSRP	Gigabit Switch Redundancy Protocol
HMAC	Keyed-Hashing for Message Authentication
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ICMPv6	Internet Control Message Protocol version 6
ID	Identifier
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers, Inc.
IETF	the Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IP	Internet Protocol
IPCP	IP Control Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPV6CP	IP Version 6 Control Protocol
IPX	Internetwork Packet Exchange
ISO	International Organization for Standardization
ISP	Internet Service Provider
IST	Internal Spanning Tree
L2LD	Layer 2 Loop Detection
LAN	Local Area Network
LCP	Link Control Protocol
LED	Light Emitting Diode
LLC	Logical Link Control
LLDP	Link Layer Discovery Protocol
LLQ+3WFQ	Low Latency Queueing + 3 Weighted Fair Queueing
LSP	Label Switched Path
LSP	Link State PDU
LSR	Label Switched Router
MA	Maintenance Association
MAC	Media Access Control
MC	Memory Card
MD5	Message Digest 5
MDI	Medium Dependent Interface
MDI-X	Medium Dependent Interface crossover
MEP	Maintenance association End Point
MIB	Management Information Base
MIP	Maintenance domain Intermediate Point
MRU	Maximum Receive Unit
MSTI	Multiple Spanning Tree Instance
MSTP	Multiple Spanning Tree Protocol
MTU	Maximum Transfer Unit
NAK	Not AcKnowledge
NAS	Network Access Server
NAT	Network Address Translation
NCP	Network Control Protocol

NDP	Neighbor Discovery Protocol
NET	Network Entity Title
NLA ID	Next-Level Aggregation Identifier
NPDU	Network Protocol Data Unit
NSAP	Network Service Access Point
NSSA	Not So Stubby Area
NTP	Network Time Protocol
OADP	Octpower Auto Discovery Protocol
OAM	Operations, Administration, and Maintenance
OSPF	Open Shortest Path First
OUI	Organizationally Unique Identifier
packet/s	packets per second (can also appear as pps)
PAD	PADding
PAE	Port Access Entity
PC	Personal Computer
PCI	Protocol Control Information
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PID	Protocol IDentifier
PIM	Protocol Independent Multicast
PIM-DM	Protocol Independent Multicast-Dense Mode
PIM-SM	Protocol Independent Multicast-Sparse Mode
PIM-SSM	Protocol Independent Multicast-Source Specific Multicast
PoE	Power over Ethernet
PRI	Primary Rate Interface
PS	Power Supply
PSNP	Partial Sequence Numbers PDU
QoS	Quality of Service
RA	Router Advertisement
RADIUS	Remote Authentication Dial In User Service
RDI	Remote Defect Indication
REJ	REJect
RFC	Request For Comments
RIP	Routing Information Protocol
RIPng	Routing Information Protocol next generation
RMON	Remote Network Monitoring MIB
RPF	Reverse Path Forwarding
RQ	ReQuest
RSTP	Rapid Spanning Tree Protocol
SA	Source Address
SD	Secure Digital
SDH	Synchronous Digital Hierarchy
SDU	Service Data Unit
SEL	NSAP SELector
SFD	Start Frame Delimiter
SFP	Small Form factor Pluggable
SFP+	Enhanced Small Form factor Pluggable
SML	Split Multi Link

SMTP	Simple Mail Transfer Protocol
SNAP	Sub-Network Access Protocol
SNMP	Simple Network Management Protocol
SNP	Sequence Numbers PDU
SNPA	Subnetwork Point of Attachment
SPF	Shortest Path First
SSAP	Source Service Access Point
STP	Spanning Tree Protocol
TA	Terminal Adapter
TACACS+	Terminal Access Controller Access Control System Plus
TCP/IP	Transmission Control Protocol/Internet Protocol
TLA ID	Top-Level Aggregation Identifier
TLV	Type, Length, and Value
TOS	Type Of Service
TPID	Tag Protocol Identifier
TTL	Time To Live
UDLD	Uni-Directional Link Detection
UDP	User Datagram Protocol
ULR	Uplink Redundant
UPC	Usage Parameter Control
UPC-RED	Usage Parameter Control - Random Early Detection
VAA	VLAN Access Agent
VLAN	Virtual LAN
VRRP	Virtual Router Redundancy Protocol
WAN	Wide Area Network
WDM	Wavelength Division Multiplexing
WFQ	Weighted Fair Queueing
WRED	Weighted Random Early Detection
WS	Work Station
WWW	World-Wide Web
XFP	10 gigabit small Form factor Pluggable

Conventions: The terms "Switch" and "switch"

The term Switch (upper-case "S") is an abbreviation for any or all of the following models:

- AX2500S series switch

The term switch (lower-case "s") might refer to a Switch, another type of switch from the current vendor, or a switch from another vendor. The context decides the meaning.

Conventions: KB, MB, GB, and TB

This manual uses the following conventions:

1 KB (kilobyte) is 1024 bytes.

1 MB (megabyte) is 1024^2 bytes.

1 GB (gigabyte) is 1024^3 bytes.

1 TB (terabyte) is 1024^4 bytes.

Contents

1. Overview of Supported MIBs

1.1 MIB system diagram

Figure 1-1 and *Figure 1-2* show the MIB system supported by this Switch.

Figure 1-1 MIB system diagram (1/2)

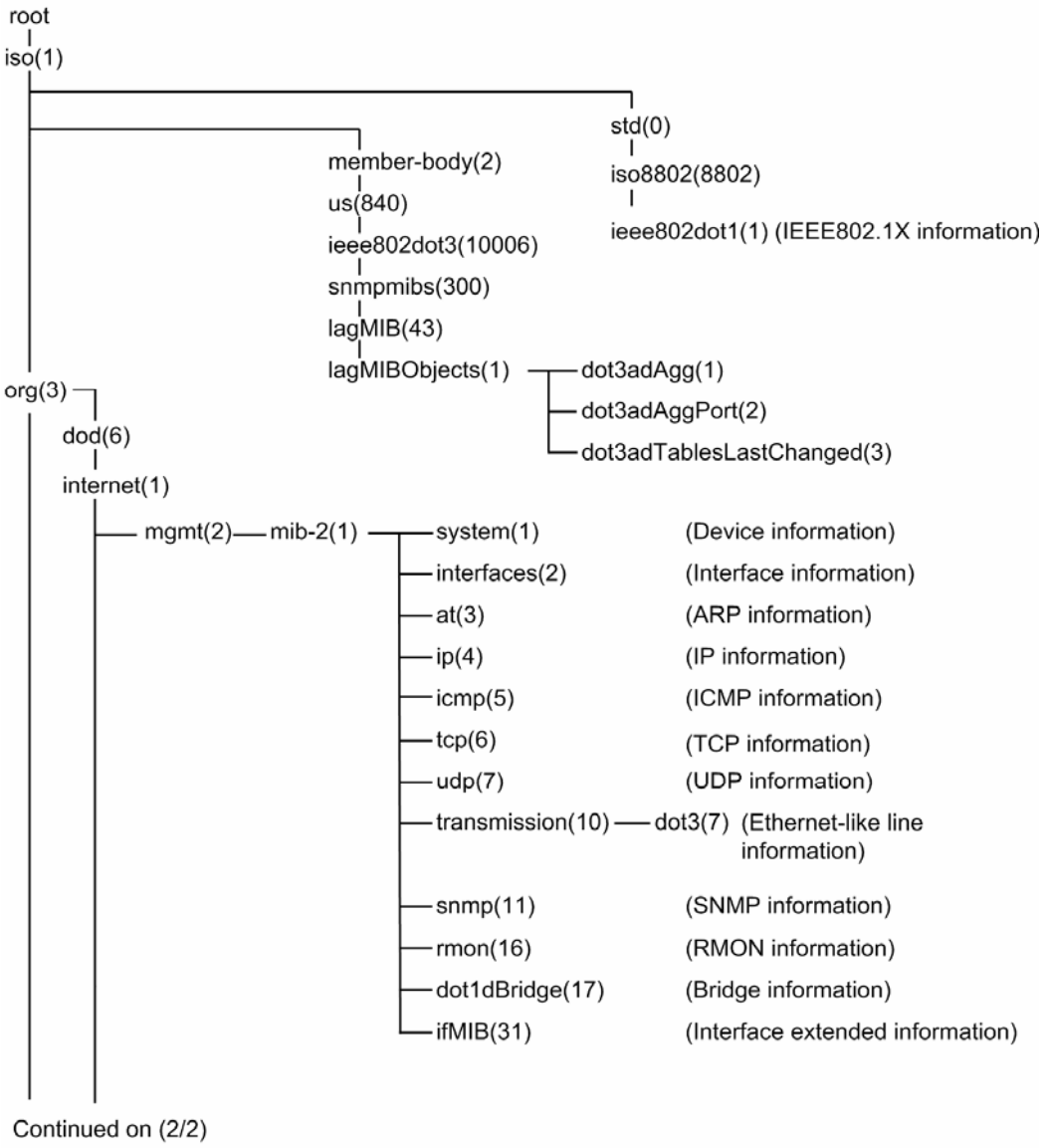
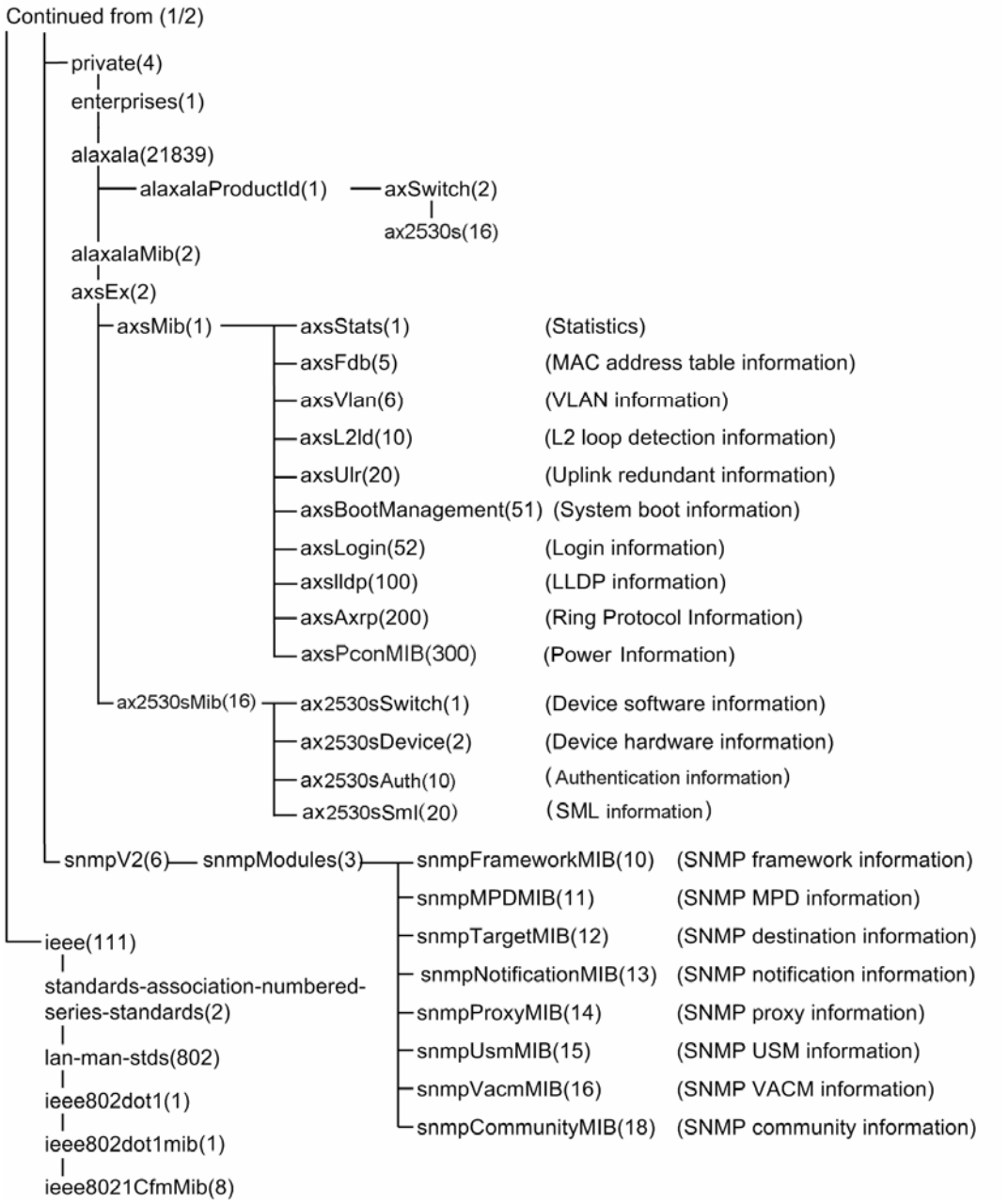


Figure 1-2 MIB system diagram (2/2)



1.2 MIB list

The following table shows the supported MIBs.

Table 1-1 MIB groups

MIB group			Functionality	Supported
Standard MIB	system group		MIB of information on a switch	Y
	interfaces group		MIB of information on an interface	Y
	at group		MIB of information on an ARP table	Y
	ip group		MIB of IP information. Includes IPv6 information.	Y
	icmp group		ICMP information MIB. Includes ICMPv6 information.	Y
	tcp group		TCP information MIB	Y
	udp group		MIB of UDP information	Y
	dot3 group		MIB of information on an Ethernet-like interface	Y
	snmp group		MIB of SNMP information	Y
	rmon group	Ethernet Statistics Group	MIB of the table on the statistics of an Ethernet interface	Y
		History Control Group	MIB of the history control table of Ethernet statistics	Y
		Ethernet History Group	MIB of the history table of Ethernet statistics	Y
		Alarm Group	MIB of information on an alarm table	Y
		Host Group	MIB of information on a host group	N
		Host Top"N"	MIB of information on a HostTopN group	N
		Matrix	MIB of information on a matrix group	N
		Filter	MIB of information on a filter group	N
		Packet Capture	MIB of information on a PacketCapture group	N
		Event Group	MIB of the table of an event generated using an RMON agent	Y

MIB group			Functionality	Supported
	dot1dBridge group	dot1dBase group	Bridge information	Y
		dot1dStp group	Information on a Spanning Tree Protocol	Y
		dot1dTp group	Forwarding information of a bridge	Y
		dot1dStatic group	Filtering information	N
		pBridgeMIB group	Priority control and multicast filtering information	Y
		qBridgeMIB group	Virtual bridge information	Y
	ifMIB group		MIB of interface extended information	Y
	ipv6MIB group		Old MIB on an IPv6 The IPv6 information for the Switch can be obtained from the ip group.	N
	ipv6IcmpMIB group		Old MIB on an ICMPv6 The ICMPv6 information for the Switch can be obtained from the icmp group.	N
	powerEthernet MIB group	pethPsePortObjects group	MIB concerning the power source characteristics of ports on the power source equipment	N
		pethMainPseObjects group	MIB concerning the attributes of the main power supply on the power source equipment	N
		pethNotificatonControl group	MIB concerning notifications from the power source equipment	N
	IEEE 8021-CFM-MIB group	dot1agCfmStack group	CFM stack table MIB	Y
		dot1agCfmDefaultMd group	CFM default MD table MIB	N
		dot1agCfmVlan group	CFM VLAN table MIB	Y
		dot1agCfmConfigErrorList group	CFM configuration information error list table MIB	N
		dot1agCfmMd group	CFM MD table MIB	Y
		dot1agCfmMaNet group	CFM MA net group MIB	Y
		dot1agCfmMaComp group	CFM MA component group MIB	Y

MIB group			Functionality	Supported
		dot1agCfmMaMepList group	CFM MA MEP list table MIB	Y
		dot1agCfmMep group	CFM MEP table MIB	Y
		dot1agCfmLtr group	CFM LTR table MIB	Y
		dot1agCfmMepDb group	CFM MEP database table MIB	Y
	IEEE 8023-LAG-MIB group	dot3adAgg group	Information on an Aggregator	Y
		dot3adAggPort group	Link Aggregation Control information on all AggregationPorts	Y
	IEEE 802.1X MIB group		MIB on IEEE 802.1X.	Y
	snmpModules group	snmpFrameworkMIB group	MIB on an SNMP framework	Y
		snmpMPDMIB group	MIB of an SNMP message and dispatcher	Y
		snmpTargetMIB group	MIB of SNMP destination information	Y
		snmpNotificationMIB group	MIB of SNMP notification information	Y
		snmpProxyMIB group	MIB of an SNMP proxy	N
		snmpUsmMIB group	MIB of an SNMP user-based security model	Y
		snmpVacmMIB group	MIB of an SNMP view-based access control model	Y
		snmpCommunityMIB group	MIB of the coexistence of SNMPv1, SNMPv2C, and SNMPv3	N
Private MIBs	axsStats group	axsIfStats group	MIB of mega unit interface statistics	Y
		axsQoS group	MIB of QoS statistics	Y
		axsDHCP group	MIB of statistics concerning the DHCP server	N
	axsGsrpMIB group	axsGsrpGroupTable group	MIB of a table that stores GSRP group information	N

MIB group			Functionality	Supported
		axsGsrpVlanGroupTable group	MIB of a table that stores GSRP VLAN group information	N
		axsGsrpNeighborGroupTable group	MIB of a table that stores the GSRP group information of an opposing switch	N
		axsGsrpNeighborVlanGroupTable group	MIB of a table that stores the GSRP VLAN group information of an opposing switch	N
	axsFdb group	axsFdbCounterTable	MIB of an information table on the MAC address table learning count	Y
	axsVlan group	axsVlanBridge group	MIB of the Bridge-MIB for each VLAN	Y
		axsVlanTagTranslation group	MIB of an information table for tag conversion	Y
	axsL2ldMIB group	axsL2ldGlobalInfo group	MIB of L2 loop detection information	Y
		axsL2ldPortTable group	MIB of a table that stores L2 loop detecting port information	Y
	axsUlr group	axsUlrGlobalInfo group	MIB of uplink redundancy information	Y
		axsUlrPortTable group	MIB of a table that stores uplink redundancy port information	Y
	axsOadp group	axsOadpGlobalInfo group	MIB of the active state of an OADP functionality	N
		axsOadpPortInfo group	MIB of OADP port information	N
		axsOadpNeighborInfo group	MIB of an OADP neighboring node	N
	axsFlow group	axsAccessFilterStats group	MIB of the table information corresponding to the number of packets that coincide with the flow detection conditions and operation information set in an access list	N
		axsQosFlowStats group	MIB of the table information corresponding to the number of packets that coincide with the flow detection conditions and operation information set in a QoS flow list	N
	axsBootManagement group		MIB on system startup	Y
	axsLogin group		MIB of logins	Y

MIB group			Functionality	Supported
	axslldp group	axslldpConfiguration group	MIB of LLDP configuration	Y
		axslldpStats group	MIB of LLDP statistics	Y
		axslldpLocalSystemData group	MIB of information on the Switch for LLDP	Y
		axslldpRemoteSystemData group	MIB of information on the neighboring switch connected with the Switch for LLDP	Y
		axslldpRemoteOriginInfoData group	MIB of ALAXALA TLV information for LLDP	Y
	axsAxrpMIB group	axsAxrpGroupTable group	MIB of Ring Protocol group information	Y
		axsAxrpVlanGroupTable group	MIB of Ring Protocol VLAN group information	Y
	axsPconMIB group	axsPconModuleData	MIB that contains a table of information about the operating status and power mode of the Switch or each board	Y
		axsPconPowerCon	MIB that contains a table of power consumption information for the entire Switch, or the consumption information for a chassis or board	Y
		axsPconTraffic	MIB that contains a table of traffic information for the Switch, or traffic information for a board	N
	ax2530sSwitch group		MIB of switch model information	Y
	ax2530sDevice	ax2530sChassis group	MIB of switch chassis information	Y
		ax2530sPhysLine group	MIB of switch interface information	Y
		ax2530sDeviceError group	MIB of switch failure information	Y
	ax2530sManagement	ax2530sFdbClearMIB group	MIB for clearing MAC address table information	N
	ax2530sAuth	ax2530sAuth group	Common MIB for authentication functionality	Y
	ax2530sSml group [OS-L2A]		Information about SML	Y
	icmp group (HP private MIB)		Private MIB for HP	N

MIB group		Functionality	Supported
	sFlow group (InMon private MIB)	Private MIB for InMon Corp.	Y

(Legend) Y: Supported in the Switch; N: Not supported in the Switch

1.3 Acquiring private MIB definition files

Private MIB definition files (ASN.1) are provided with the software.

1.4 Format of MIB descriptions

The format of the descriptions for supported MIBs described in this manual is given below. For each MIB, the identifiers and implementation specifications are described for each group.

- **ID**

  This is the certified format for descriptions of object IDs.

  Example: The format and object ID value of the identifier for the private MIB axsStats group are shown below.

```
Identifier:      axsStats OBJECT IDENTIFIER ::= {axsMib 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.1
```

  For details about the object ID value of a private MIB, see *A Private MIB Names and Object ID Values*.

- **Implementation specifications**

  The table below provides the implementation specifications for each MIB.

  The implementation specifications of the axsStats group are used as an example to explain the private MIB items. The following table shows the sample implementation specifications for the axsIfStats group.

Table 1-2 axsStats Group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsIfStatsTable {axsIfStats 1}	NOT-ACCESSIBLE	NA	Extended statistics table of the interface	Y
2	axsIfStatsEntry {axsIfStatsTable 1}	NOT-ACCESSIBLE	NA	Extended statistics table entry of the interface. INDEX {axsIfStatsIndex}	Y
3	axsIfStatsIndex {axsIfStatsEntry 1}	NOT-ACCESSIBLE	NA	Interface index of the Switch. Same as ifIndex.	Y
. . .	. . .	. . .	. . .	. . .	. . .

Object identifier

This indicates the object identifier name of the MIB.

SYNTAX

The meaning of SYNTAX used in a private MIB is described in the table below. Syntax is described only in the implementation specifications of a private MIB.

Table 1-3 Meaning of SYNTAX elements used in a private MIB

#	SYNTAX	Description of SYNTAX
1	Counter	Integer value that increases from 0 to 4294967295 ($2^{32}-1$) and returns to 0.

#	SYNTAX	Description of SYNTAX
2	Counter32	Integer value that increases from 0 to 4294967295 ($2^{32}-1$) and returns to 0.
3	Counter64	Integer value that increases from 0 to 18446744073709551615 ($2^{64}-1$) and returns to 0.
4	DisplayString	String of 0 or more and 255 or fewer characters (each byte is an NVT ASCII value).
5	Gauge	Integer of 0 or greater, whose value can be increased and decreased (in the range from 0 to 4294967295).
6	INTEGER	Indicates integer information in the range from -2147483648 to 2147483647 (from -2^{31} to $2^{31}-1$).
7	Integer32	Indicates integer information in the range from -2147483648 to 2147483647 (from -2^{31} to $2^{31}-1$).
8	OCTET STRING	String of 0 or more characters (in 8-bit units). Each byte is from 0 to 255.
9	IpAddress	4-byte OCTET STRING (that contains a 32-bit IP address).
10	Ipv6Address	16-byte OCTET STRING (that contains a 128-bit IPv6 address).
11	OBJECT IDENTIFIER	Stores the sequence-fixed list of a sub-ID.
12	MacAddress	802 MAC address, indicated in regular order, as defined by IEEE 802.1a. OCTET STRING type.
13	RowStatus	SYNTAX type for controlling the generation or deletion of a conceptual line entry.
14	TimeStamp	Time stamp for measuring the time from a certain event in units of 1/100 seconds.
15	TimeTicks	Positive integer. The time from a certain event is indicated in units of 1/100 seconds.
16	BITS	String of named bits represented by a string of 0 or more characters (in 8-bit units). "1" is assigned to the corresponding bit. Named bit 0 corresponds to the most significant bit, and the logical sum of each named bit is returned by GetResponse. (Example) The value returned by GetResponse becomes 0x82 when named bits 0 and 6 are valid.
17	NOT-ACCESSIBLE	Cannot be accessed.
18	PortList	Bitmap of a port. "1" is assigned to the bit corresponding to a valid port. In this Switch, "1" is assigned to the bit corresponding to the ifIndex number of a port.
19	VlanIndex	Indicates the index number (1 to 4094) of a VLAN.
20	AddressFamilyNumbers	Address number to which IANA is assigned
21	VlanIdOrZero	Indicates the index number (1 to 4094) of a VLAN.

#	SYNTAX	Description of SYNTAX
22	SnmpAdminString	Character string including management information. DisplayString type.
23	InetAddressType	Internet address type
24	InetAddress	Internet address. OCTET STRING type.
25	OwnerString	String containing from 0 to 127 characters. Indicates the owner name of the resource assigned in management. DisplayString type.
26	BridgeId	Bridge ID used in a Spanning Tree Protocol. OCTET STRING type.
27	Timeout	STP timer with units of 1/100 seconds
28	TruthValue	True or false value
29	InterfaceIndex	ifIndex number managed by the system. Integer value in the range from 1 to 2147483647 (2^{31}-1)
30	Unsigned32	Integer information in the range from 0 to 4294967295 (2^{32}-1)
31	TimeFilter	Positive integer. Index number for indicating the time from a certain event in units of 1/100 seconds.

Access

- R/O: Indicates that the MIB access in a standard document is Read_Only.
- R/W: Indicates that the MIB access in a standard document is Read_Write.
- R/NW: Indicates that the MIB access in a standard document is Read_Write, but in this Switch it is Read_Only.
- R/C: Indicates that the MIB access in a standard document is Read_Create.
- R/NC: Indicates that the MIB access in a standard document is Read_Create, but in this Switch it is Read_Only.
- AN: Indicates that the MIB access in a standard document is accessible-for-notify. An object cannot be acquired and set, but AN can be read as the variable of a trap.
- NA: Indicates that the MIB access in a standard document is not-accessible.

Implementation specifications

[Standard]: Describes the standard overview of a standard document.

[Implementation]: Describes the implementation specifications in the Switch.

Implemented

- Y: Indicates a MIB that is supported in (responds to) the Switch. However, the MIB does not respond if the Access column is NA. Note that a response varies depending on the functionality used.

- M: Indicates a MIB that is supported in (responds to) the Switch. However, the Switch always responds with a fixed value because it cannot use a statistics counter to count.
- N: Indicates a MIB that is not supported in (does not respond to) the Switch.

2. Standard MIBs

This chapter describes the implementation specifications for the standard MIBs used by the Switch.

2.1 system group (MIB-II)

(1) ID

```
system OBJECT IDENTIFIER ::= {mib-2 1}
Object ID value: 1.3.6.1.2.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the system group.

Table 2-1 system group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	sysDescr {system 1}	R/O	[Standard] The name or version number of a hardware switch, operating system, or network operating system. [Implementation] A character string made up of a company name, switch type, switch model, software name, software type name, software version, and software abbreviation. (Example) For the AX2530S-24T: `ALAXALA AX2530 AX-xxxx-xxx-x [AX25xxS-xxx] Switching software Ver. 3.0 [OS-xxx]` `AX-xxxx-xxx-x`: Device part name `AX25xxS-xxx`: Device model `Switching Software`: Software name `Ver. 3.0`: Software version `OS-xxx`: Software abbreviation	Y
2	sysObjectID {system 2}	R/O	[Standard] The authentication ID of a network management subsystem vendor. [Implementation] A fixed value. For AX2530S: 1.3.6.1.4.1.21839.1.2.16	Y
3	sysUpTime {system 3}	R/O	[Standard] The time elapsed since the system was started (10 millisecond intervals). [Implementation] The time elapsed since the switch was started.	Y
4	sysContact {system 4}	R/NW	[Standard] A contact for a management node. [Implementation] A character string of no more than 60 characters specified by using a configuration command. The default is empty (`NULL`).	Y
5	sysName {system 5}	R/NW	[Standard] The name or domain name of the management node. [Implementation] A character string of no more than 60 characters specified by using a configuration command. The default is empty (`NULL`).	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	sysLocation {system 6}	R/NW	[Standard] The location on which the management node is installed. [Implementation] A character string of no more than 60 characters specified by using a configuration command. The default is empty (NULL).	Y
7	sysServices {system 7}	R/O	[Standard] A value indicating the service. [Implementation] Fixed value of 2.	Y

2.2 interfaces group (MIB-II)

2.2.1 interfaces group

(1) ID

```
interfaces OBJECT IDENTIFIER ::= {mib-2 2}
Object ID value: 1.3.6.1.2.1.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the interfaces group.

Table 2-2 interfaces group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	ifNumber {interfaces 1}	R/O	[Standard] The number of network interfaces provided by this system. [Implementation] Same as the standard. Changing the interface configuration causes the value of this object to change.	Y
2	ifTable {interfaces 2}	NA	[Standard] A table of interface entities. [Implementation] Same as the standard.	Y
3	ifEntry {ifTable 1}	NA	[Standard] A list of interfaces that belong to the sub-network layer. INDEX { ifIndex } [Implementation] Same as the standard.	Y
4	ifIndex {ifEntry 1}	R/O	[Standard] The number that identifies the interface. The value ranges from 1 to ifNumber. [Implementation] Changing the interface configuration causes the value of this object to change. ifIndex is assigned as follows: [24T] [24T4X] [24S4X] [24TD] [24S4XD] ● Port: 10 + port number − 1 ● Link aggregation: 60 + channel group number ● VLAN: 200 + VLAN ID ● Peer-link: 5 [48T] [48T2X] [48TD] ● Port: 10 + port number − 1 ● Link aggregation: 62 + channel group number ● VLAN: 200 + VLAN ID ● Peer-link: 5	Y
5	ifDescr {ifEntry 2}	R/O	[Standard] Interface information. [Implementation] A fixed character string and a set-up character string for each interface type.	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	ifType {ifEntry 3}	R/O	[Standard] The interface type. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Ethernet-csmacd (6) ● When the ifIndex indicates a VLAN: l2vlan (135) ● When the ifIndex indicates a link aggregation: ieee8023adLag (161) ● When the ifIndex indicates a peer-link: other (1).	Y
7	ifMtu {ifEntry 4}	R/O	[Standard] The maximum size of datagrams that can be sent and received by this interface (octets). [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: The MTU (for an Ethernet interface belonging to the VLAN), the system MTU information, or the IP MTU information (only during setup), whichever is the smallest. ● When the ifIndex indicates a link aggregation: The smallest MTU value of an Ethernet interface that belongs to link aggregation. ● When the ifIndex indicates a peer-link: Same as the standard.	Y
8	ifSpeed {ifEntry 5}	R/O	[Standard] The current line speed (bit/s) of this interface. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: If no bandwidth is set by using a configuration command, the line speed of the interface in question is shown. If a bandwidth has been set, the set bandwidth is shown. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Show the sum of the line speed values of the physical ports used by the link aggregation. ● When the ifIndex indicates a peer-link: The sum of the line speed values of the physical ports used by the peer-link.	Y

#	Object identifier	Access	Implementation specifications	Implemented
9	ifPhysAddress {ifEntry 6}	R/O	[Standard] The physical address directly below the network layer of this interface. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Returns a response (in canonical form) of the value that represents the MAC address. ● When the ifIndex indicates a VLAN: Returns a response (in canonical form) of the value that represents the MAC address assigned to the VLAN. ● When the ifIndex indicates a link aggregation: Returns a response (in canonical form) of the value that represents the MAC address of the channel group. ● When the ifIndex indicates a peer-link: Fixed value of `00 00 00 00 00 00`.	Y
10	ifAdminStatus {ifEntry 7}	R/W	[Standard] The desired status of this interface: ● up (1) ● down (2) ● testing (3) [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: If `shutdown` is specified in the configuration, the status is specified as down (2). ● When the ifIndex indicates a VLAN: If `vlan suspend` is specified in the configuration, the status is specified as down (2). ● When the ifIndex indicates a link aggregation: I `shutdown` is specified for the channel group in the configuration, the status is specified as down (2). ● When the ifIndex indicates a peer-link: Fixed value of up (1). `Read_Only`.	Y
11	ifOperStatus {ifEntry 8}	R/O	[Standard] The current status of this interface: ● up (1) ● down (2) ● testing (3) [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Same as the standard. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: down (2) for Standalone or Conflict; up (1) for Full.	Y

#	Object identifier	Access	Implementation specifications	Implemented
12	ifLastChange {ifEntry 9}	R/O	[Standard] sysUpTime when the ifOperStatus of this interface last changed (in units of 1/100 seconds). [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Same as the standard. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Same as the standard.	Y
13	ifInOctets {ifEntry 10}	R/O	[Standard] The number of octets received by this interface. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The received number of octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The received number of octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
14	ifInUcastPkts {ifEntry 11}	R/O	[Standard] The number of unicast packets sent to the upper protocol. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
15	ifInNUcastPkts {ifEntry 12}	R/O	[Standard] The number of non-unicast packets (broadcast and multicast packets) sent to the upper layer protocol. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
16	ifInDiscards {ifEntry 13}	R/O	[Standard] The number of packets that contained no errors but failed to be sent to the upper layer protocol (or the number of packets received but discarded, for example, because there was no buffer). [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The number of events discarded due to a receive FIFO overflow. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The number of events discarded due to a receive FIFO overflow. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
17	ifInErrors {ifEntry 14}	R/O	[Standard] The number of packets discarded because they contained errors. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The number of packets discarded because they contained errors such as an FCS error, short packet, maximum packet length error, collided packets, invalid packet format, or odd bit error. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The number of packets discarded because they contained errors such as an FCS error, short packet, maximum packet length error, collided packets, invalid packet format, or odd bit error. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
18	ifInUnknownProtos {ifEntry 15}	R/O	[Standard] The number of packets received but discarded because the protocol is unsupported. [Implementation] ● Fixed value of 0. ● When the ifIndex indicates a peer-link: Fixed value of 0.	M
19	ifOutOctets {ifEntry 16}	R/O	[Standard] The number of octets of packets sent by this interface. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The transmitted number of octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The transmitted number of octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
20	ifOutUcastPkts {ifEntry 17}	R/O	[Standard] The number of unicast packets sent by the upper layer. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
21	ifOutNUcastPkts {ifEntry 18}	R/O	[Standard] The number of non-unicast packets sent by the upper layer. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The number of normal non-unicast packets sent by the upper layer. (The number of packets whose MAC DA contains an I/G bit of 1. However, this does not include MAC packets, but includes SMT.) ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The number of normal non-unicast packets sent by the upper layer. (The number of packets whose the MAC DA contains an I/G bit of 1. However, this does not include MAC packets, but includes SMT.) ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
22	ifOutDiscards {ifEntry 19}	R/O	[Standard] The number of packets that contained no errors but were discarded (for example, due to insufficient send buffers) before being sent. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: The number of events discarded due to a send FIFO overflow (underrun). ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The number of events discarded due to a send FIFO overflow (underrun). ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
23	ifOutErrors {ifEntry 20}	R/O	[Standard] The number of packets that failed to be sent due to an error [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
24	ifOutQLen {ifEntry 21}	R/O	[Standard] The size of the send packet queue. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: The sum of the send packet queue sizes of the ports that belong to the channel group. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
25	ifSpecific {ifEntry 22}	R/O	[Standard] The reference to a MIB that defines the medium features of the interface. This is the object ID of an ifType-dependent MIB. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Returns a response of `1.3.6.1.2.1.10.7` when normal. Otherwise `0.0`. ● When the ifIndex indicates a VLAN: Fixed value of `0.0`. ● When the ifIndex indicates a link aggregation: Fixed value of `0.0`. ● When the ifIndex indicates a peer-link: Fixed value of `0.0`.	Y

2.3 at group (MIB-II)

(1) ID

```
at OBJECT IDENTIFIER ::= {mib-2 3}
Object ID value: 1.3.6.1.2.1.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the at group.

Table 2-3 at group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	atTable {at 1}	NA	[Standard] An address translation table that contains the mappings of the NetworkAddresses to the equivalent physical addresses. Some interfaces do not use the translation table to determine the equivalent addresses. For these interfaces, the address translation table is empty and contains no entries. [Implementation] Same as the standard. Indicates the mappings of network addresses to physical addresses.	Y
2	atEntry {atTable 1}	NA	[Standard] Each entry represents the mapping of a single NetworkAddress to its equivalent physical address. INDEX {atIfIndex,atNetAddress } [Implementation] Same as the standard.	Y
3	atIfIndex {atEntry 1}	R/NW	[Standard] The ifIndex value of the corresponding interface. [Implementation] The ifIndex of the interface that has an atPhysAddress. `Read_Only`.	Y
4	atPhysAddress {atEntry 2}	R/NW	[Standard] The physical address. [Implementation] The MAC address dependent on the ARP table, which is medium-dependent.	Y
5	atNetAddress {atEntry 3}	R/NW	[Standard] The IP address corresponding to the medium-dependent atPhysAddress. [Implementation] Same as the standard.	Y

2.4 ip group (MIB-II)

The ip group (MIB-II) is described in the following documents:

- RFC 1213 (March 1991)
- RFC 2011 (November 1996)
- RFC 4293 (April 2006)

2.4.1 ip

(1) ID

```
ip OBJECT IDENTIFIER ::= {mib-2 4}
Object ID value: 1.3.6.1.2.1.4
ipTrafficStats OBJECT IDENTIFIER ::= {ip 31}
Object ID value: 1.3.6.1.2.1.4.31
```

(2) Implementation specifications

The following table shows the implementation specifications for the ip group.

Table 2-4 ip group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	ipForwarding {ip 1}	R/NW	[Standard] Indicates whether IP forwarding is available (or indicates whether the object serves as a gateway): ● gateway/forwarding (1) ● host/notforwarding (2) [Implementation] Fixed value of host/notforwarding (2). Read_Only.	Y
2	ipDefaultTTL {ip 2}	R/NW	[Standard] A default value specified for the TTL of the IP header. [Implementation] Same as the standard. Read_Only.	Y
3	ipInReceives {ip 3}	R/O	[Standard] The total number of IP datagrams received across all the interfaces. [Implementation] Same as the standard. ● Fixed value of 0.	M
4	ipInHdrErrors {ip 4}	R/O	[Standard] The number of datagrams received but discarded due to an IP header error. [Implementation] Includes the number of IP packets that contain an error, such as an IP header checksum error, version error, header length error (TTL over), or format error. ● Fixed value of 0.	M
5	ipInAddrErrors {ip 5}	R/O	[Standard] The number of packets discarded because the IP header contained an invalid destination address. [Implementation] Runs a count when the class of the destination address is not A, B, C, or D. It also runs a count when the destination is an invalid broadcast address (255.255.255.255 or 0.0.0.0).	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	ipForwDatagrams {ip 6}	R/O	[Standard] The number of packets determined to be forwarded. [Implementation] Same as the standard. ● Fixed value of 0.	Y
7	ipInUnknownProtos {ip 7}	R/O	[Standard] The number of IP datagrams discarded because an unknown or unsupported protocol was detected in an incoming IP packet that was received. [Implementation] Same as the standard.	Y
8	ipInDiscards {ip 8}	R/O	[Standard] The total number of sent IP datagrams discarded due to a reason other than an error. [Implementation] The number of packets received when ifOperStatus is not up.	Y
9	ipInDelivers {ip 9}	R/O	[Standard] The number of IP datagrams sent to the upper layer. [Implementation] The number of incoming IP packets received.	Y
10	ipOutRequests {ip 10}	R/O	[Standard] The total number of IP datagrams requested to be sent as IP packets by the upper layer. [Implementation] The number of outgoing IP packets sent.	Y
11	ipOutDiscards {ip 11}	R/O	[Standard] The number of IP datagrams discarded due to a reason other than an error. [Implementation] The number of IP packets discarded prior to being sent due to congestion control or insufficient send buffers. ● Fixed value of 0.	Y
12	ipOutNoRoutes {ip 12}	R/O	[Standard] The number of IP datagrams discarded because the transmission route was not specified. [Implementation] Runs a count before packet forwarding when the destination network is not found in the routing table.	Y
13	ipReasmTimeout {ip 13}	R/O	[Standard] The maximum number of seconds to hold a fragmented packet waiting to be reassembled. [Implementation] Same as the standard.	Y
14	ipReasmReqds {ip 14}	R/O	[Standard] The number of received IP datagrams to be reassembled. [Implementation] Same as the standard.	Y
15	ipReasmOKs {ip 15}	R/O	[Standard] The number of received IP datagrams that were successfully reassembled. [Implementation] Same as the standard.	Y
16	ipReasmFails {ip 16}	R/O	[Standard] The number of received IP datagrams that failed to be reassembled. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
17	ipFragOKs {ip 17}	R/O	[Standard] The number of IP datagrams that successfully fragmented. [Implementation] Same as the standard.	Y
18	ipFragFails {ip 18}	R/O	[Standard] The number of IP datagrams that failed to fragment. [Implementation] Runs a count when the required fragmentation of a packet cannot be performed because the DF bit of the IP header is on. A count is also run when a buffer for fragmentation cannot be reserved.	Y
19	ipFragCreates {ip 19}	R/O	[Standard] The number of IP datagram fragments generated by fragmentation. [Implementation] Same as the standard.	Y
20	ipAddrTable {ip 20}	NA	[Standard] A table of addressing information related to the IP addresses for this entity (addressing information table that is IP address-specific). [Implementation] Same as the standard.	Y
21	ipAddrEntry {ipAddrTable 1}	NA	[Standard] A list of addressing information related to one of the IP addresses for this entity. [Implementation] Same as the standard.	Y
22	ipAdEntAddr {ipAddrEntry 1}	R/O	[Standard] An IP address. [Implementation] Port IP address.	Y
23	ipAdEntIfIndex {ipAddrEntry 2}	R/O	[Standard] The interface index value applied by this entry. Same value as ifIndex. [Implementation] Same as the standard.	Y
24	ipAdEntNetMask {ipAddrEntry 3}	R/O	[Standard] The subnet mask for the IP address of this entry. [Implementation] Same as the standard.	Y
25	ipAdEntBcastAddr {ipAddrEntry 4}	R/O	[Standard] The lowest bit value of the address used when sending an IP broadcast. [Implementation] Same as the standard.	Y
26	ipAdEntReasmMaxSize {ipAddrEntry 5}	R/O	[Standard] The maximum size of the IP packets created by reassembling the IP datagrams received and fragmented by the interface. [Implementation] Same as the standard.	Y
27	ipRouteTable {ip 21}	NA	[Standard] Route table. [Implementation] Not implemented.	N
28	ipNetToMediaTable {ip 22}	NA	[Standard] An IP address translation table used to map the IP addresses to the physical addresses. [Implementation] Same as the standard.	Y
29	ipNetToMediaEntry {ipNetToMediaTable 1}	NA	[Standard] A list of IP addresses. Each IP address is associated with a physical address. INDEX {ipNetToMediaIfIndex, ipNetToMediaNetAddress} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
30	ipNetToMediaIfIndex {ipNetToMediaEntry 1}	R/NC	[Standard] The validated interface ID number. [Implementation] Same as the standard. Read_Only.	Y
31	ipNetToMediaPhysAddress {ipNetToMediaEntry 2}	R/NC	[Standard] The medium-dependent physical address. [Implementation] Same as the standard. Read_Only.	Y
32	ipNetToMediaNetAddress {ipNetToMediaEntry 3}	R/NC	[Standard] An IP address that corresponds to a medium-dependent IP address. [Implementation] Same as the standard. Read_Only.	Y
33	ipNetToMediaType {ipNetToMediaEntry 4}	R/NC	[Standard] The mapping type: ● other (1) ● invalid (2) ● dynamic (3) ● static (4) [Implementation] Same as the standard. Read_Only.	Y
34	ipRoutingDiscards {ip 23}	R/O	[Standard] The number of routing entries that are valid but selected to be discarded. For example, this indicates the number of entries discarded due to insufficient routing buffers. [Implementation] Fixed value of 0.	Y
35	ipv6IpForwarding {ip 25}	R/NW	[Standard] Whether to forward IPv6 packets sent to other devices. ● forwarding (1) ● notForwarding (2) [Implementation] Same as the standard. Read_Only.	Y
36	ipv6IpDefaultHopLimit {ip 26}	R/NW	[Standard] Default value of HopLimit for IPv6. [Implementation] Same as the standard. Read_Only.	Y
37	ipv4InterfaceTableLastChange {ip 27}	R/O	[Standard] sysUpTime value when ipv4InterfaceTable last changed. [Implementation] Same as the standard.	Y
38	ipv4InterfaceTable {ip 28}	NA	[Standard] IPv4 information for each interface. [Implementation] Same as the standard.	Y
39	ipv4InterfaceEntry {ipv4InterfaceTable 1}	NA	[Standard] IPv4 information related to a specific interface. INDEX {ipv4InterfaceIfIndex} [Implementation] Same as the standard.	Y
40	ipv4InterfaceIfIndex {ipv4InterfaceEntry 1}	NA	[Standard] ifIndex value for identifying an IP interface. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
41	ipv4InterfaceReasmMaxSize {ipv4InterfaceEntry 2}	R/O	[Standard] Maximum size for assembling fragments. [Implementation] Same as the standard.	Y
42	ipv4InterfaceEnableStatus {ipv4InterfaceEntry 3}	R/NW	[Standard] Whether IPv4 is in operation in this interface. ● up (1) ● down (2) [Implementation] Same as the standard. Read_Only.	Y
43	ipv4InterfaceRetransmitTime {ipv4InterfaceEntry 4}	R/O	[Standard] Interval for re-sending ARP (in milliseconds). [Implementation] Same as the standard.	Y
44	ipv6InterfaceTableLastChange {ip 29}	R/O	[Standard] sysUpTime value when ipv6InterfaceTable last changed. [Implementation] Same as the standard.	Y
45	ipv6InterfaceTable {ip 30}	NA	[Standard] IPv6 information for each interface. [Implementation] Same as the standard.	Y
46	ipv6InterfaceEntry {ipv6InterfaceTable 1}	NA	[Standard] IPv6 information related to a specific interface. INDEX {ipv6InterfaceIfIndex} [Implementation] Same as the standard.	Y
47	ipv6InterfaceIfIndex {ipv6InterfaceEntry 1}	NA	[Standard] ifIndex value to identify an IPv6 interface. [Implementation] Same as the standard.	Y
48	ipv6InterfaceReasmMaxSize {ipv6InterfaceEntry 2}	R/O	[Standard] Maximum size for assembling fragments. [Implementation] Same as the standard.	Y
49	ipv6InterfaceIdentifier {ipv6InterfaceEntry 3}	R/O	[Standard] Interface identifier. [Implementation] Same as the standard.	Y
50	ipv6InterfaceEnableStatus {ipv6InterfaceEntry 5}	R/NW	[Standard] Whether IPv6 is in operation in this interface. ● up (1) ● down (2) [Implementation] Same as the standard. Read_Only.	Y
51	ipv6InterfaceReachableTime {ipv6InterfaceEntry 6}	R/O	[Standard] Expected time in which the neighbor is able to arrive (in milliseconds). [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
52	ipv6InterfaceRetransmitTime {ipv6InterfaceEntry 7}	R/O	[Standard] Interval of the NDP to be re-sent (in milliseconds). [Implementation] Same as the standard.	Y
53	ipv6InterfaceForwarding {ipv6InterfaceEntry 8}	R/NW	[Standard] Whether to forward the received IPv6 packets sent to other devices. ● forwarding (1) ● notForwarding (2) [Implementation] Same as the standard. Read_Only.	Y
54	ipSystemStatsTable {ipTrafficStats 1}	NA	[Standard] Total statistics of devices for each IP version. [Implementation] Same as the standard.	Y
55	ipSystemStatsEntry {ipSystemStatsTable 1}	NA	[Standard] Total statistics of devices related to a specific IP version. INDEX {ipSystemStatsIPVersion} [Implementation] Same as the standard.	Y
56	ipSystemStatsIPVersion {ipSystemStatsEntry 1}	NA	[Standard] IP version. [Implementation] Same as the standard.	Y
57	ipSystemStatsInReceives {ipSystemStatsEntry 3}	R/O	[Standard] The number of IP packets received. [Implementation] Same as the standard.	Y
58	ipSystemStatsHCInReceives {ipSystemStatsEntry 4}	R/O	[Standard] The number of IP packets received (64-bit). [Implementation] Same as the standard.	Y
59	ipSystemStatsInOctets {ipSystemStatsEntry 5}	R/O	[Standard] The total number of octets of IP packets received. [Implementation] Same as the standard.	Y
60	ipSystemStatsHCInOctets {ipSystemStatsEntry 6}	R/O	[Standard] The total number of octets of IP packets received (64-bit). [Implementation] Same as the standard.	Y
61	ipSystemStatsInHdrErrors {ipSystemStatsEntry 7}	R/O	[Standard] The number of incoming packets discarded due to an invalid header. [Implementation] Same as the standard.	Y
62	ipSystemStatsInNoRoutes {ipSystemStatsEntry 8}	R/O	[Standard] The number of incoming packets discarded due to lack of a route. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
63	ipSystemStatsInAddrErrors {ipSystemStatsEntry 9}	R/O	[Standard] The number of incoming packets discarded due to an invalid address. [Implementation] Same as the standard.	Y
64	ipSystemStatsInUnknownProtos {ipSystemStatsEntry 10}	R/O	[Standard] The number of incoming packets discarded due to an unsupported protocol. [Implementation] Same as the standard.	Y
65	ipSystemStatsInTruncatedPkts {ipSystemStatsEntry 11}	R/O	[Standard] The number of incoming packets discarded due to insufficient packet length. [Implementation] Same as the standard.	Y
66	ipSystemStatsInForwDatagrams {ipSystemStatsEntry 12}	R/O	[Standard] The number of forwarded IP packets. [Implementation] Same as the standard.	Y
67	ipSystemStatsHCInForwDatagrams {ipSystemStatsEntry 13}	R/O	[Standard] The number of forwarded IP packets (64-bit). [Implementation] Same as the standard.	Y
68	ipSystemStatsReasmReqds {ipSystemStatsEntry 14}	R/O	[Standard] The number of received fragments that need to be assembled. [Implementation] Same as the standard.	Y
69	ipSystemStatsReasmOKs {ipSystemStatsEntry 15}	R/O	[Standard] The number of fragments that were assembled successfully. [Implementation] Same as the standard.	Y
70	ipSystemStatsReasmFails {ipSystemStatsEntry 16}	R/O	[Standard] The number of fragments for which assembly failed. [Implementation] Same as the standard.	Y
71	ipSystemStatsInDiscards {ipSystemStatsEntry 17}	R/O	[Standard] The number of incoming packets that were discarded for no reason that could be found in the packet itself. [Implementation] Same as the standard.	Y
72	ipSystemStatsInDelivers {ipSystemStatsEntry 18}	R/O	[Standard] The number of incoming packets that were delivered to the upper level (including ICMP). [Implementation] Same as the standard.	Y
73	ipSystemStatsHCInDelivers {ipSystemStatsEntry 19}	R/O	[Standard] The number of incoming packets that were delivered to the upper level (including ICMP) (64-bit). [Implementation] Same as the standard.	Y
74	ipSystemStatsOutRequests {ipSystemStatsEntry 20}	R/O	[Standard] The number of outgoing packets that were delivered from the upper level (including ICMP). [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
75	ipSystemStatsHCOutRequests {ipSystemStatsEntry 21}	R/O	[Standard] The number of outgoing packets that were delivered from the upper level (including ICMP) (64-bit). [Implementation] Same as the standard.	Y
76	ipSystemStatsOutNoRoutes {ipSystemStatsEntry 22}	R/O	[Standard] The number of packets sent from this device and discarded due to lack of a route. [Implementation] Same as the standard.	Y
77	ipSystemStatsOutForwDatagrams {ipSystemStatsEntry 23}	R/O	[Standard] The number of outgoing packets by forwarding. [Implementation] Same as the standard.	Y
78	ipSystemStatsHCOutForwDatagrams {ipSystemStatsEntry 24}	R/O	[Standard] The number of outgoing packets by forwarding (64-bit). [Implementation] Same as the standard.	Y
79	ipSystemStatsOutDiscards {ipSystemStatsEntry 25}	R/O	[Standard] The number of outgoing packets that were discarded. [Implementation] Same as the standard.	Y
80	ipSystemStatsOutFragReqds {ipSystemStatsEntry 26}	R/O	[Standard] The number of outgoing packets that need to be split into fragments. [Implementation] Same as the standard.	Y
81	ipSystemStatsOutFragOKs {ipSystemStatsEntry 27}	R/O	[Standard] The number of time a packet was split into fragments successfully. [Implementation] Same as the standard.	Y
82	ipSystemStatsOutFragFails {ipSystemStatsEntry 28}	R/O	[Standard] The number of outgoing packets discarded due to a failure to split the packet into fragments. [Implementation] Same as the standard.	Y
83	ipSystemStatsOutFragCreates {ipSystemStatsEntry 29}	R/O	[Standard] The number of fragments that were generated due to splitting outgoing packets. [Implementation] Same as the standard.	Y
84	ipSystemStatsOutTransmits {ipSystemStatsEntry 30}	R/O	[Standard] The number of outgoing IP packets. [Implementation] Same as the standard.	Y
85	ipSystemStatsHCOutTransmits {ipSystemStatsEntry 31}	R/O	[Standard] The number of outgoing IP packets (64-bit). [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
86	ipSystemStatsOutOctets {ipSystemStatsEntry 32}	R/O	[Standard] The number of octets for outgoing IP packets. [Implementation] Same as the standard.	Y
87	ipSystemStatsHCOutOctets {ipSystemStatsEntry 33}	R/O	[Standard] The number of octets for outgoing IP packets (64-bit). [Implementation] Same as the standard.	Y
88	ipSystemStatsInMcastPkts {ipSystemStatsEntry 34}	R/O	[Standard] The number of IP packets for incoming multicasts. [Implementation] Same as the standard.	Y
89	ipSystemStatsHCInMcastPkts {ipSystemStatsEntry 35}	R/O	[Standard] The number of IP packets for incoming multicasts (64-bit). [Implementation] Same as the standard.	Y
90	ipSystemStatsInMcastOctets {ipSystemStatsEntry 36}	R/O	[Standard] The total number of octets for incoming multicast IP packets. [Implementation] Same as the standard.	Y
91	ipSystemStatsHCInMcastOctets {ipSystemStatsEntry 37}	R/O	[Standard] The total number of octets for incoming multicast IP packets (64-bit). [Implementation] Same as the standard.	Y
92	ipSystemStatsOutMcastPkts {ipSystemStatsEntry 38}	R/O	[Standard] The number of IP packets for outgoing multicasts. [Implementation] Same as the standard.	Y
93	ipSystemStatsHCOutMcastPkts {ipSystemStatsEntry 39}	R/O	[Standard] The number of IP packets for outgoing multicasts (64-bit). [Implementation] Same as the standard.	Y
94	ipSystemStatsOutMcastOctets {ipSystemStatsEntry 40}	R/O	[Standard] The total number of octets for outgoing multicast IP packets. [Implementation] Same as the standard.	Y
95	ipSystemStatsHCOutMcastOctets {ipSystemStatsEntry 41}	R/O	[Standard] The total number of octets for outgoing multicast IP packets (64-bit). [Implementation] Same as the standard.	Y
96	ipSystemStatsInBcastPkts {ipSystemStatsEntry 42}	R/O	[Standard] The number of IP packets for received broadcasts. [Implementation] Same as the standard.	Y
97	ipSystemStatsHCInBcastPkts {ipSystemStatsEntry 43}	R/O	[Standard] The number of IP packets for received broadcasts (64-bit). [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
98	ipSystemStatsOutBcastPkts {ipSystemStatsEntry 44}	R/O	[Standard] The number of IP packets for sent broadcasts. [Implementation] Same as the standard.	Y
99	ipSystemStatsHCOutBcastPkts {ipSystemStatsEntry 45}	R/O	[Standard] The number of IP packets for sent broadcasts (64-bit). [Implementation] Same as the standard.	Y
100	ipSystemStatsDiscontinuityTime {ipSystemStatsEntry 46}	R/O	[Standard] sysUpTime value when there is a discontinuity in the statistics at the end. [Implementation] Same as the standard.	Y
101	ipSystemStatsRefreshRate {ipSystemStatsEntry 47}	R/O	[Standard] Valid polling interval (in milliseconds). [Implementation] Same as the standard.	Y
102	ipIfStatsTableLastChange {ipTrafficStats 2}	R/O	[Standard] sysUpTime value when the last row of ipIfStatsTable has been added or deleted. [Implementation] Not implemented.	N
103	ipIfStatsTable {ipTrafficStats 3}	NA	[Standard] Statistics for each IP version and interface. [Implementation] Not implemented.	N
104	ipAddressPrefixTable {ip 32}	NA	[Standard] Prefixes for references from other objects. [Implementation] Same as the standard.	Y
105	ipAddressPrefixEntry {ipAddressPrefixTable 1}	NA	[Standard] ipAddressPrefixTable entries. INDEX {ipAddressPrefixIfIndex, ipAddressPrefixType,ipAddressPrefixPrefix,ipAddressPrefixLength} [Implementation] Same as the standard.	Y
106	ipAddressPrefixIfIndex {ipAddressPrefixEntry 1}	NA	[Standard] ifIndex for the interface. [Implementation] Same as the standard.	Y
107	ipAddressPrefixType {ipAddressPrefixEntry 2}	NA	[Standard] Address type. [Implementation] Same as the standard.	Y
108	ipAddressPrefixPrefix {ipAddressPrefixEntry 3}	NA	[Standard] Address prefix. [Implementation] Same as the standard.	Y
109	ipAddressPrefixLength {ipAddressPrefixEntry 4}	NA	[Standard] Address prefix length. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
110	ipAddressPrefixOrigin {ipAddressPrefixEntry 5}	R/O	[Standard] Prefix origin. [Implementation] Same as the standard.	Y
111	ipAddressPrefixOnLinkFlag {ipAddressPrefixEntry 6}	R/O	[Standard] Whether the prefix can be used for on-link checks. [Implementation] Same as the standard.	Y
112	ipAddressPrefixAutonomousFlag {ipAddressPrefixEntry 7}	R/O	[Standard] Whether the prefix can be used for auto generation of IPv6 addresses. [Implementation] Same as the standard.	Y
113	ipAddressPrefixAdvPreferredLifetime {ipAddressPrefixEntry 8}	R/O	[Standard] Time to stay in the `preferred` status (in seconds). [Implementation] Same as the standard.	Y
114	ipAddressPrefixAdvValidLifetime {ipAddressPrefixEntry 9}	R/O	[Standard] Valid time (in seconds). [Implementation] Same as the standard.	Y
115	ipAddressSpinLock {ip 33}	R/NW	[Standard] Spin lock. [Implementation] Indefinite value.	M
116	ipAddressTable {ip 34}	NA	[Standard] Correspondence table for IP addresses and interfaces. [Implementation] Same as the standard.	Y
117	ipAddressEntry {ipAddressTable 1}	NA	[Standard] ipAddressTable entries. INDEX { ipAddressAddrType, ipAddressAddr } [Implementation] Same as the standard.	Y
118	ipAddressAddrType {ipAddressEntry 1}	NA	[Standard] Address type. [Implementation] Same as the standard.	Y
119	ipAddressAddr {ipAddressEntry 2}	NA	[Standard] Address. [Implementation] Same as the standard.	Y
120	ipAddressIfIndex {ipAddressEntry 3}	R/NC	[Standard] ifIndex value. [Implementation] Same as the standard. `Read_Only`.	Y
121	ipAddressType {ipAddressEntry 4}	R/NC	[Standard] Address type. ● unicast (1) ● anycast (2) ● broadcast (3) [Implementation] Same as the standard. `Read_Only`.	Y

#	Object identifier	Access	Implementation specifications	Implemented
122	ipAddressPrefix {ipAddressEntry 5}	R/O	[Standard] Prefix. [Implementation] Same as the standard.	Y
123	ipAddressOrigin {ipAddressEntry 6}	R/O	[Standard] Address origin. [Implementation] Same as the standard.	Y
124	ipAddressStatus {ipAddressEntry 7}	R/NC	[Standard] Address status. [Implementation] Same as the standard. Read_Only.	Y
125	ipAddressCreated {ipAddressEntry 8}	R/O	[Standard] sysUpTime value when the address was generated. [Implementation] Same as the standard.	Y
126	ipAddressLastChanged {ipAddressEntry 9}	R/O	[Standard] sysUpTime value when the entry was last updated. [Implementation] Same as the standard.	Y
127	ipAddressRowStatus {ipAddressEntry 10}	R/NC	[Standard] Entry status. [Implementation] Same as the standard. Read_Only.	Y
128	ipAddressStorageType {ipAddressEntry 11}	R/NC	[Standard] Entry storage type. [Implementation] Same as the standard. Read_Only.	Y
129	ipNetToPhysicalTable {ip 35}	NA	[Standard] Correspondence table for net addresses and physical addresses. [Implementation] Same as the standard.	Y
130	ipNetToPhysicalEntry {ipNetToPhysicalTable 1}	NA	[Standard] ipNetToPhysicalTable entries. INDEX {ipNetToPhysicalIfIndex,ipNetToPhysicalNetAddressType,ipNetToPhysicalNetAddress} [Implementation] Same as the standard.	Y
131	ipNetToPhysicalIfIndex {ipNetToPhysicalEntry 1}	NA	[Standard] Ifindex of the interface. [Implementation] Same as the standard.	Y
132	ipNetToPhysicalNetAddressType {ipNetToPhysicalEntry 2}	NA	[Standard] Net address type. [Implementation] Same as the standard.	Y
133	ipNetToPhysicalNetAddress {ipNetToPhysicalEntry 3}	NA	[Standard] Net address. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
134	ipNetToPhysicalPhysAddress {ipNetToPhysicalEntry 4}	R/NC	[Standard] Physical address. [Implementation] Same as the standard. Read_Only.	Y
135	ipNetToPhysicalLastUpdated {ipNetToPhysicalEntry 5}	R/O	[Standard] sysUpTime value when the entry was last updated. [Implementation] Same as the standard.	Y
136	ipNetToPhysicalType {ipNetToPhysicalEntry 6}	R/NC	[Standard] Physical address type. ● other (1) ● invalid (2) ● dynamic (3) ● static (4) ● local (5) [Implementation] Same as the standard. Read_Only.	Y
137	ipNetToPhysicalState {ipNetToPhysicalEntry 7}	R/O	[Standard] Physical address status. ● reachable (1) ● stale (2) ● delay (3) ● probe (4) ● invalid (5) ● unknown (6) ● incomplete (7) [Implementation] Same as the standard.	Y
138	ipNetToPhysicalRowStatus {ipNetToPhysicalEntry 8}	R/NC	[Standard] Entry status. [Implementation] Same as the standard. Read_Only.	Y
139	ipv6ScopeZoneIndexTable {ip 36}	NA	[Standard] Table for IPv6 scope. [Implementation] Same as the standard.	Y
140	ipv6ScopeZoneIndexEntry {ipv6ScopeZoneIndexTable 1}	NA	[Standard] Zone for a specific interface. INDEX {ipv6ScopeZoneIndexIfIndex} [Implementation] Same as the standard.	Y
141	ipv6ScopeZoneIndexIfIndex {ipv6ScopeZoneIndexEntry 1}	NA	[Standard] ifIndex value for the interface. [Implementation] Same as the standard.	Y
142	ipv6ScopeZoneIndexLinkLocal {ipv6ScopeZoneIndexEntry 2}	R/O	[Standard] Scope zone for link local. [Implementation] Same as the standard.	Y
143	ipv6ScopeZoneIndex3 {ipv6ScopeZoneIndexEntry 3}	R/O	[Standard] Scope zone for index 3. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
144	ipv6ScopeZoneIndexAdminLocal {ipv6ScopeZoneIndexEntry 4}	R/O	[Standard] Scope zone for local management. [Implementation] Same as the standard.	Y
145	ipv6ScopeZoneIndexSiteLocal {ipv6ScopeZoneIndexEntry 5}	R/O	[Standard] Scope zone for local site. [Implementation] Same as the standard.	Y
146	ipv6ScopeZoneIndex6 {ipv6ScopeZoneIndexEntry 6}	R/O	[Standard] Scope zone for index 6. [Implementation] Same as the standard.	Y
147	ipv6ScopeZoneIndex7 {ipv6ScopeZoneIndexEntry 7}	R/O	[Standard] Scope zone for index 7. [Implementation] Same as the standard.	Y
148	ipv6ScopeZoneIndexOrganizationLocal {ipv6ScopeZoneIndexEntry 8}	R/O	[Standard] Scope zone for local organization. [Implementation] Same as the standard.	Y
149	ipv6ScopeZoneIndex9 {ipv6ScopeZoneIndexEntry 9}	R/O	[Standard] Scope zone for index 9. [Implementation] Same as the standard.	Y
150	ipv6ScopeZoneIndexA {ipv6ScopeZoneIndexEntry 10}	R/O	[Standard] Scope zone for index A. [Implementation] Same as the standard.	Y
151	ipv6ScopeZoneIndexB {ipv6ScopeZoneIndexEntry 11}	R/O	[Standard] Scope zone for index B. [Implementation] Same as the standard.	Y
152	ipv6ScopeZoneIndexC {ipv6ScopeZoneIndexEntry 12}	R/O	[Standard] Scope zone for index C. [Implementation] Same as the standard.	Y
153	ipv6ScopeZoneIndexD {ipv6ScopeZoneIndexEntry 13}	R/O	[Standard] Scope zone for index D. [Implementation] Same as the standard.	Y
154	ipDefaultRouterTable {ip 37}	NA	[Standard] Table for the default router. [Implementation] Same as the standard. For IPv6 only.	Y

#	Object identifier	Access	Implementation specifications	Implemented
155	ipDefaultRouterEntry {ipDefaultRouterTable 1}	NA	[Standard] ipDefaultRouterTable entries. INDEX {ipDefaultRouterAddressType,ipDefaultRouterAddress,ipDefaultRouterIfIndex} [Implementation] Same as the standard.	Y
156	ipDefaultRouterAddressType {ipDefaultRouterEntry 1}	NA	[Standard] Address type. [Implementation] Same as the standard.	Y
157	ipDefaultRouterAddress {ipDefaultRouterEntry 2}	NA	[Standard] Address for the default router. [Implementation] Same as the standard.	Y
158	ipDefaultRouterIfIndex {ipDefaultRouterEntry 3}	NA	[Standard] ifIndex of the interface that connects with the default router. [Implementation] Same as the standard.	Y
159	ipDefaultRouterLifetime {ipDefaultRouterEntry 4}	R/O	[Standard] Valid time (in seconds). [Implementation] Same as the standard.	Y
160	ipDefaultRouterPreference {ipDefaultRouterEntry 5}	R/O	[Standard] Priority. ● reserved (-2) ● low (-1) ● medium (0) ● high (1) [Implementation] Fixed value of 0.	Y
161	ipv6RouterAdvertSpinLock {ip 38}	R/W	[Standard] Spin lock of ipv6RouterAdvertTable. [Implementation] Not implemented.	N
162	ipv6RouterAdvertTable {ip 39}	NA	[Standard] Table for generating RA. [Implementation] Not implemented.	N

2.5 icmp group (MIB-II)

The icmp group (MIB-II) is described in the following documents:

- RFC 1213 (March)
- RFC 2011 (November 1996)
- RFC 4293 (April 2006)

(1) ID

```
icmp OBJECT IDENTIFIER ::= {mib-2 5}
Object ID value: 1.3.6.1.2.1.5
```

(2) Implementation specifications

The following table shows the implementation specifications for the icmp group.

Table 2-5 icmp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	icmpInMsgs {icmp 1}	R/O	[Standard] The total number of ICMP messages received by this entity. [Implementation] Same as the standard.	Y
2	icmpInErrors {icmp 2}	R/O	[Standard] The number of ICMP message errors received (such as checksum errors and frame length errors). [Implementation] Same as the standard.	Y
3	icmpInDestUnreachs {icmp 3}	R/O	[Standard] The number of ICMP Destination Unreachable messages received. [Implementation] Same as the standard.	Y
4	icmpInTimeExcds {icmp 4}	R/O	[Standard] The number of ICMP Time Exceeded messages received. [Implementation] Same as the standard.	Y
5	icmpInParmProbs {icmp 5}	R/O	[Standard] The number of ICMP Parameter Problem messages received. [Implementation] Same as the standard.	Y
6	icmpInSrcQuenchs {icmp 6}	R/O	[Standard] The number of ICMP Source Quench messages received. [Implementation] Same as the standard.	Y
7	icmpInRedirects {icmp 7}	R/O	[Standard] The number of ICMP Network Redirect messages received. [Implementation] Same as the standard.	Y
8	icmpInEchos {icmp 8}	R/O	[Standard] The number of ICMP Echo request messages received. [Implementation] Same as the standard.	Y
9	icmpInEchoReps {icmp 9}	R/O	[Standard] The number of ICMP Echo reply messages received. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
10	icmpInTimestamps {icmp 10}	R/O	[Standard] The number of ICMP Timestamp request messages received. [Implementation] Same as the standard.	Y
11	icmpInTimestampReps {icmp 11}	R/O	[Standard] The number of ICMP Timestamp reply messages received. [Implementation] Same as the standard.	Y
12	icmpInAddrMasks {icmp 12}	R/O	[Standard] The number of ICMP Address Mask request messages received. [Implementation] Same as the standard.	Y
13	icmpInAddrMaskReps {icmp 13}	R/O	[Standard] The number of ICMP Address Mask reply messages received. [Implementation] Same as the standard.	Y
14	icmpOutMsgs {icmp 14}	R/O	[Standard] The total number of attempts to send ICMP messages (including error attempts). [Implementation] Same as the standard.	Y
15	icmpOutErrors {icmp 15}	R/O	[Standard] The number of ICMP messages that failed to be sent due to an error. [Implementation] Only runs a count when there are no buffers.	Y
16	icmpOutDestUnreachs {icmp 16}	R/O	[Standard] The number of ICMP Destination Unreachable messages sent. [Implementation] Same as the standard.	Y
17	icmpOutTimeExcds {icmp 17}	R/O	[Standard] The number of ICMP Time Exceeded messages sent. [Implementation] Same as the standard.	Y
18	icmpOutParmProbs {icmp 18}	R/O	[Standard] The number of ICMP Parameter Problem messages sent. [Implementation] Same as the standard.	Y
19	icmpOutSrcQuenchs {icmp 19}	R/O	[Standard] The number of ICMP Source Quench messages sent. [Implementation] Same as the standard.	Y
20	icmpOutRedirects {icmp 20}	R/O	[Standard] The number of ICMP Redirect messages sent. [Implementation] Same as the standard.	Y
21	icmpOutEchos {icmp 21}	R/O	[Standard] The number of ICMP Echo request messages sent. [Implementation] Same as the standard.	Y
22	icmpOutEchoReps {icmp 22}	R/O	[Standard] The number of ICMP Echo reply messages sent. [Implementation] Same as the standard.	Y
23	icmpOutTimestamps {icmp 23}	R/O	[Standard] The number of ICMP Timestamp request messages sent. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
24	icmpOutTimestampReps {icmp 24}	R/O	[Standard] The number of ICMP Timestamp reply messages sent. [Implementation] Same as the standard.	Y
25	icmpOutAddrMasks {icmp 25}	R/O	[Standard] The number of ICMP Address Mask request messages sent. [Implementation] Same as the standard.	Y
26	icmpOutAddrMaskReps {icmp 26}	R/O	[Standard] The number of ICMP Address Mask reply messages sent. [Implementation] Same as the standard.	Y
27	icmpStatsTable {icmp 29}	NA	[Standard] ICMP statistics for each IP version and the total system. [Implementation] Same as the standard.	Y
28	icmpStatsEntry {icmpStatsTable 1}	NA	[Standard] icmpStatsTable entries. INDEX {icmpStatsIPVersion} [Implementation] Same as the standard.	Y
29	icmpStatsIPVersion {icmpStatsEntry 1}	NA	[Standard] IP version. [Implementation] Same as the standard.	Y
30	icmpStatsInMsgs {icmpStatsEntry 2}	R/O	[Standard] The number of ICMP messages received. [Implementation] Same as the standard.	Y
31	icmpStatsInErrors {icmpStatsEntry 3}	R/O	[Standard] The number of errors received (such as an unknown ICMP check sum). [Implementation] Same as the standard.	Y
32	icmpStatsOutMsgs {icmpStatsEntry 4}	R/O	[Standard] The number of ICMP messages sent. [Implementation] Same as the standard.	Y
33	icmpStatsOutErrors {icmpStatsEntry 5}	R/O	[Standard] The number of errors sent (such as an insufficient buffer). [Implementation] Same as the standard.	Y
34	icmpMsgStatsTable {icmp 30}	NA	[Standard] Statistics for each ICMP type. [Implementation] Same as the standard.	Y
35	icmpMsgStatsEntry {icmpMsgStatsTable 1}	NA	[Standard] icmpMsgStatsTable entries. INDEX {icmpMsgStatsIPVersion, icmpMsgStatsType} [Implementation] Same as the standard.	Y
36	icmpMsgStatsIPVersion {icmpMsgStatsEntry 1}	NA	[Standard] IP version. [Implementation] Same as the standard.	Y
37	icmpMsgStatsType {icmpMsgStatsEntry 2}	NA	[Standard] ICMP type. [Implementation] Same as the standard.	Y
38	icmpMsgStatsInPkts {icmpMsgStatsEntry 3}	R/O	[Standard] The number of messages received. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
39	icmpMsgStatsOutPkts {icmpMsgStatsEntry 4}	R/O	[Standard] The number of messages sent. [Implementation] Same as the standard.	Y

2.6 tcp group (MIB-II)

The tcp group is described in the following documents:

- RFC 1213 (March 1991)
- RFC 2012 (November 1996)
- RFC 4022 (March 2005)

2.6.1 tcp

(1) ID

```
tcp OBJECT IDENTIFIER ::= {mib-2 6}
Object ID value: 1.3.6.1.2.1.6
```

(2) Implementation specifications

The following table shows the implementation specifications for the tcp group.

Table 2-6 tcp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	tcpRtoAlgorithm {tcp 1}	R/O	[Standard] The algorithm used to determine the timeout for resending: ● other (1) ● constant (2) ● rsre (3) ● vanj (4) ● rfc2988 (5) [Implementation] Same as the standard.	Y
2	tcpRtoMin {tcp 2}	R/O	[Standard] The minimum resend timeout value (in milliseconds). [Implementation] Same as the standard.	Y
3	tcpRtoMax {tcp 3}	R/O	[Standard] The maximum resend timeout value (in milliseconds). [Implementation] Same as the standard.	Y
4	tcpMaxConn {tcp 4}	R/O	[Standard] The total number of TCP connections that can be supported. If the maximum number of connections is dynamic, a response of -1 is returned. [Implementation] Same as the standard.	Y
5	tcpActiveOpens {tcp 5}	R/O	[Standard] The number of TCP connection state transitions from CLOSE to SYN-SENT. [Implementation] Same as the standard.	Y
6	tcpPassiveOpens {tcp 6}	R/O	[Standard] The number of TCP connection state transitions from LISTEN to SYN-RCVD. [Implementation] Same as the standard.	Y
7	tcpAttemptFails {tcp 7}	R/O	[Standard] The sum of the number of TCP connection state transitions from SYN-SENT or SYN-RCVD to CLOSE, and the number from SYN-RCVD to LISTEN. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
8	tcpEstabResets {tcp 8}	R/O	[Standard] The number of TCP connection state transitions from ESTABLISHED or CLOSE-WAIT to CLOSE. [Implementation] Same as the standard.	Y
9	tcpCurrEstab {tcp 9}	R/O	[Standard] The total number of TCP connections in the ESTABLISHED or CLOSE-WAIT state. [Implementation] Same as the standard.	Y
10	tcpInSegs {tcp 10}	R/O	[Standard] The total number of segments received that contain an erroneous segment. [Implementation] Same as the standard.	Y
11	tcpOutSegs {tcp 11}	R/O	[Standard] The total number of segments sent. [Implementation] Same as the standard.	Y
12	tcpRetransSegs {tcp 12}	R/O	[Standard] The total number of segments resent. [Implementation] Same as the standard.	Y
13	tcpConnTable {tcp 13}	NA	[Standard] A table that contains information specific to the TCP connections. [Implementation] Same as the standard.	Y
14	tcpConnEntry {tcpConnTable 1}	NA	[Standard] Entry information for a specific TCP connection. INDEX {tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort} [Implementation] Same as the standard.	Y
15	tcpConnState {tcpConnEntry 1}	R/NW	[Standard] The TCP connection state. ● closed (1) ● listen (2) ● synSent (3) ● synReceived (4) ● established (5) ● finWait1 (6) ● finWait2 (7) ● closeWait (8) ● lastAck (9) ● closing (10) ● timeWait (11) ● deleteTCB (12) [Implementation] Same as the standard. Read_Only.	Y
16	tcpConnLocalAddress {tcpConnEntry 2}	R/O	[Standard] The local IP address for this TCP connection. [Implementation] Same as the standard.	Y
17	tcpConnLocalPort {tcpConnEntry 3}	R/O	[Standard] The local port number for this TCP connection. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
18	tcpConnRemAddress {tcpConnEntry 4}	R/O	[Standard] The remote IP address for this TCP connection. [Implementation] Same as the standard.	Y
19	tcpConnRemPort {tcpConnEntry 5}	R/O	[Standard] The remote port number for this TCP connection. [Implementation] Same as the standard.	Y
20	tcpInErrs {tcp 14}	R/O	[Standard] The total number of erroneous segments received. [Implementation] Same as the standard.	Y
21	tcpOutRsts {tcp 15}	R/O	[Standard] The number of segments sent that have an RST flag. [Implementation] Same as the standard.	Y
22	tcpHCInSegs {tcp 17}	R/O	[Standard] The total number of segments received including error segments (64-bit). [Implementation] Not implemented.	N
23	tcpHCOutSegs {tcp 18}	R/O	[Standard] The total number of segments sent (64-bit). [Implementation] Not implemented.	N
24	tcpConnectionTable {tcp 19}	NA	[Standard] A table that lists TCP connection information. [Implementation] Same as the standard.	Y
25	tcpConnectionEntry {tcpConnectionTable 1}	NA	[Standard] Entry information related to a specific TCP connection. INDEX {tcpConnectionLocalAddressType,tcpConnectionLocalAddress,tcpConnectionLocalPort,tcpConnectionRemAddressType,tcpConnectionRemAddress,tcpConnectionRemPort } [Implementation] Same as the standard.	Y
26	tcpConnectionLocalAddressType {tcpConnectionEntry 1}	NA	[Standard] Local IP address type for this TCP connection. [Implementation] Same as the standard.	Y
27	tcpConnectionLocalAddress {tcpConnectionEntry 2}	NA	[Standard] Local IP address for this TCP connection. [Implementation] Same as the standard.	Y
28	tcpConnectionLocalPort {tcpConnectionEntry 3}	NA	[Standard] Local port number for this TCP connection. [Implementation] Same as the standard.	Y
29	tcpConnectionRemAddressType {tcpConnectionEntry 4}	NA	[Standard] Remote IP address type for this TCP connection. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
30	tcpConnectionRemAddress {tcpConnectionEntry 5}	NA	[Standard] Remote IP address for this TCP connection. [Implementation] Same as the standard.	Y
31	tcpConnectionRemPort {tcpConnectionEntry 6}	NA	[Standard] Remote port number for this TCP connection. [Implementation] Same as the standard.	Y
32	tcpConnectionState {tcpConnectionEntry 7}	R/NW	[Standard] TCP connection status. ● closed (1) ● listen (2) ● synSent (3) ● synReceived (4) ● established (5) ● finWait1 (6) ● finWait2 (7) ● closeWait (8) ● lastAck (9) ● closing (10) ● timeWait (11) ● deleteTCB (12) [Implementation] Same as the standard. Read_Only.	Y
33	tcpConnectionProcess {tcpConnectionEntry 8}	R/O	[Standard] TCP connection process. [Implementation] Same as the standard.	Y
34	tcpListenerTable {tcp 20}	NA	[Standard] A table that lists TCP listener information. [Implementation] Same as the standard.	Y
35	tcpListenerEntry {tcpListenerTable 1}	NA	[Standard] Entry information related to a specific TCP listener. INDEX {tcpListenerLocalAddressType,tcpListenerLocalAddress,tcpListenerLocalPort } [Implementation] Same as the standard.	Y
36	tcpListenerLocalAddressType {tcpListenerEntry 1}	NA	[Standard] Local IP address type for this TCP listener. [Implementation] Same as the standard.	Y
37	tcpListenerLocalAddress {tcpListenerEntry 2}	NA	[Standard] Local IP address for this TCP listener. [Implementation] Same as the standard.	Y
38	tcpListenerLocalPort {tcpListenerEntry 3}	NA	[Standard] Local port number for this TCP listener. [Implementation] Same as the standard.	Y
39	tcpListenerProcess {tcpListenerEntry 4}	R/O	[Standard TCP listener process. [Implementation] Same as the standard.	Y

2.7 udp group (MIB-II)

The udp group is described in the following documents:

- RFC 1213 (March 1991)
- RFC 2013 (November 1996)
- RFC 4113 (June 2005)

2.7.1 udp

(1) ID

```
udp OBJECT IDENTIFIER ::= {mib-2 7}
Object ID value: 1.3.6.1.2.1.7
```

(2) Implementation specifications

The following table shows the implementation specifications for the udp group.

Table 2-7 udp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	udpInDatagrams {udp 1}	R/O	[Standard] The number of UDP datagrams sent to the upper layer. [Implementation] Same as the standard.	Y
2	udpNoPorts {udp 2}	R/O	[Standard] The total number of UDP datagrams received for which there is no upper application on a destination port. [Implementation] Same as the standard.	Y
3	udpInErrors {udp 3}	R/O	[Standard] The number of UDP datagrams that failed to be sent to the application due to reasons other than udpNoPorts. [Implementation] Same as the standard.	Y
4	udpOutDatagrams {udp 4}	R/O	[Standard] The total number of UDP datagrams sent by the upper application. [Implementation] Same as the standard.	Y
5	udpTable {udp 5}	NA	[Standard] A table that lists UDP listener information. [Implementation] Same as the standard.	Y
6	udpEntry {udpTable 1}	NA	[Standard] The number of entries for a specific UDP listener. INDEX {udpLocalAddress,udpLocalPort} [Implementation] Same as the standard.	Y
7	udpLocalAddress {udpEntry 1}	R/O	[Standard] The local IP address for this UDP listener. [Implementation] Same as the standard.	Y
8	udpLocalPort {udpEntry 2}	R/O	[Standard] The local port number for this UDP listener. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
9	udpEndpointTable {udp 7}	NA	[Standard] A table that lists UDP endpoint information. [Implementation] Same as the standard.	Y
10	udpEndpointEntry {udpEndpointTable 1}	NA	[Standard] Entries related to a specific UDP end point. INDEX {udpEndpointLocalAddressType,udpEndpointLocalAddress,udpEndpointLocalPort,udpEndpointRemoteAddressType,udpEndpointRemoteAddress,udpEndpointRemotePort,udpEndpointInstance} [Implementation] Same as the standard.	Y
11	udpEndpointLocalAddressType {udpEndpointEntry 1}	NA	[Standard] The local IP address type for this UDP end point. [Implementation] Same as the standard.	Y
12	udpEndpointLocalAddress {udpEndpointEntry 2}	NA	[Standard] The local IP address for this UDP end point. [Implementation] Same as the standard.	Y
13	udpEndpointLocalPort {udpEndpointEntry 3}	NA	[Standard] The local port number for this UDP end point. [Implementation] Same as the standard.	Y
14	udpEndpointRemoteAddressType {udpEndpointEntry 4}	NA	[Standard] The remote IP address type for this UDP end point. [Implementation] Same as the standard.	Y
15	udpEndpointRemoteAddress {udpEndpointEntry 5}	NA	[Standard] The remote IP address for this UDP end point. [Implementation] Same as the standard.	Y
16	udpEndpointRemotePort {udpEndpointEntry 6}	NA	[Standard] The remote port number for this UDP end point. [Implementation] Same as the standard.	Y
17	udpEndpointInstance {udpEndpointEntry 7}	NA	[Standard] The instance for this UDP end point. [Implementation] Same as the standard.	Y
18	udpEndpointProcess {udpEndpointEntry 8}	R/O	[Standard] The process for this UDP end point. [Implementation] Same as the standard.	Y
19	udpHCInDatagrams {udp 8}	R/O	[Standard] The number of UDP datagrams that were reported to an upper layer (64-bit). [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
20	udpHCOutDatagrams {udp 9}	R/O	[Standard] The total number of UDP datagrams that were sent from an upper layer (64-bit). [Implementation] Not implemented.	N

2.8 dot3 group (Ethernet-Like MIB)

The dot3 group is described in the following document:

- RFC 1643

(1) ID

```
dot3 OBJECT IDENTIFER ::= {transmission 7}
Object ID value: 1.3.6.1.2.1.10.7
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot3 group.

Table 2-8 dot3 group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot3StatsTable {dot3 2}	NA	[Standard] A table of statistics for Ethernet-like interfaces connected to a specific system. [Implementation] Same as the standard.	Y
2	dot3StatsEntry {dot3StatsTable 1}	NA	[Standard] A list of statistics for a specific interface to an Ethernet-like medium. INDEX {dot3StatsIndex} [Implementation] Same as the standard.	Y
3	dot3StatsIndex {dot3StatsEntry 1}	R/O	[Standard] The index value of the interface to an Ethernet-like medium. [Implementation] Same as the standard.	Y
4	dot3StatsAlignmentErrors {dot3StatsEntry 2}	R/O	[Standard] The number of frames received that are detected by FCS checking and have an incorrect frame length.# [Implementation] Same as the standard.	M
5	dot3StatsFCSErrors {dot3StatsEntry 3}	R/O	[Standard] The number of frames received that are detected by FCS checking and have the correct frame length.# [Implementation] The number of frames received that are detected by FCS checking and have the correct frame length. If 10GBASE-R is used, the long frames received are also included in the count.	Y
6	dot3StatsSingleCollisionFrames {dot3StatsEntry 4}	R/O	[Standard] The number of frames sent successfully that only encountered a single collision. [Implementation] Same as the standard.	Y
7	dot3StatsMultipleCollisionFrames {dot3StatsEntry 5}	R/O	[Standard] The number of frames sent successfully by a specific interface that encountered two or more collisions. [Implementation] Same as the standard.	Y
8	dot3StatsSQETestErrors {dot3StatsEntry 6}	R/O	[Standard] The number of SQE TEST ERROR messages encountered. [Implementation] Fixed value of 0.	M

#	Object identifier	Access	Implementation specifications	Implemented
9	dot3StatsDeferredTransmissions {dot3StatsEntry 7}	R/O	[Standard] The number of frames whose initial transmission was delayed because the transmission line was busy. [Implementation] Same as the standard.	Y
10	dot3StatsLateCollisions {dot3StatsEntry 8}	R/O	[Standard] The number of collisions detected after a 512-bit time elapsed. [Implementation] Same as the standard.	Y
11	dot3StatsExcessiveCollisions {dot3StatsEntry 9}	R/O	[Standard] The number of transfer failures due to excessive collisions (16 times) [Implementation] Same as the standard.	Y
12	dot3StatsInternalMacTransmitErrors {dot3StatsEntry 10}	R/O	[Standard] The number of send failures due to a send error in the MAC sublayer. [Implementation] Fixed value of 0.	M
13	dot3StatsCarrierSenseErrors {dot3StatsEntry 11}	R/O	[Standard] The number of no-carrier errors that occurred during transmission [Implementation] Same as the standard.	Y
14	dot3StatsFrameTooLongs {dot3StatsEntry 13}	R/O	[Standard] The number of received frames that exceed the maximum allowable frame length.# [Implementation] Same as the standard. (Includes Alignment Error for 100BASE-TX ports.)	Y
15	dot3StatsInternalMacReceiveErrors {dot3StatsEntry 16}	R/O	[Standard] The number of frames that failed to be received due to a reception error in the MAC sublayer. [Implementation] Fixed value of 0.	M
16	dot3StatsEtherChipSet {dot3StatsEntry 17}	R/O	[Standard] The object identifier of the chipset used in the interface. [Implementation] Fixed value of 0.0.	M

#:

The frame length indicates the length from the MAC header to the FCS field. For details about the frame format, see *14.1.3 Control on the MAC and LLC sublayers* in the manual *Configuration Guide Vol. 1*.

2.9 snmp group (MIB-II)

(1) ID

```
snmp OBJECT IDENTIFIER ::= {mib-2 11}
Object ID value: 1.3.6.1.2.1.11
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmp group.

Table 2-9 snmp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	snmpInPkts {snmp 1}	R/O	[Standard] The total number of SNMP received messages. [Implementation] Same as the standard.	Y
2	snmpOutPkts {snmp 2}	R/O	[Standard] The total number of SNMP sent messages. [Implementation] Same as the standard.	Y
3	snmpInBadVersions {snmp 3}	R/O	[Standard] The total number of messages received from an unsupported version. [Implementation] Same as the standard.	Y
4	snmpInBadCommunityNames {snmp 4}	R/O	[Standard] The total number of SNMP received messages from an unused community. [Implementation] Same as the standard. SNMP Version 3 is excluded.	Y
5	snmpInBadCommunityUses {snmp 5}	R/O	[Standard] The total number of messages received that indicated a non-permitted operation in the community. [Implementation] Same as the standard. SNMP Version 3 is excluded.	Y
6	snmpInASNParseErrs {snmp 6}	R/O	[Standard] The total number of messages received that have an ASN.1 error. [Implementation] Same as the standard.	Y
7	snmpInBadTypes {snmp 7}	R/O	[Standard] The total number of unknown PDU types that are received. [Implementation] Fixed value of 0.	Y
8	snmpInTooBigs {snmp 8}	R/O	[Standard] The total number of received PDUs that have an error status of tooBig. [Implementation] Same as the standard. Fixed value of 0.	Y
9	snmpInNoSuchNames {snmp 9}	R/O	[Standard] The total number of received PDUs that have an error status of noSuchName. [Implementation] Same as the standard. Fixed value of 0.	Y
10	snmpInBadValues {snmp 10}	R/O	[Standard] The total number of received PDUs that have an error status of badValue. [Implementation] Same as the standard. Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
11	snmpInReadOnlys {snmp 11}	R/O	[Standard] The total number of received PDUs that have an error status of readOnly. [Implementation] Same as the standard. Fixed value of 0.	Y
12	snmpInGenErrs {snmp 12}	R/O	[Standard] The total number of received PDUs that have an error status of genErr. [Implementation] Same as the standard. Fixed value of 0.	Y
13	snmpInTotalReqVars {snmp 13}	R/O	[Standard] The total number of MIB objects for which a MIB was successfully collected. [Implementation] Same as the standard.	Y
14	snmpInTotalSetVars {snmp 14}	R/O	[Standard] The total number of MIB objects for which a MIB was successfully configured. [Implementation] Same as the standard.	Y
15	snmpInGetRequests {snmp 15}	R/O	[Standard] The total number of GetRequestPDUs received. [Implementation] Same as the standard.	Y
16	snmpInGetNexts {snmp 16}	R/O	[Standard] The total number of GetNextRequestPDUs received. [Implementation] Same as the standard.	Y
17	snmpInSetRequests {snmp 17}	R/O	[Standard] The total number of SetRequestPDUs received. [Implementation] Same as the standard.	Y
18	snmpInGetResponses {snmp 18}	R/O	[Standard] The total number of GetResponsePDUs received. [Implementation] Same as the standard. Fixed value of 0.	Y
19	snmpInTraps {snmp 19}	R/O	[Standard] The total number of Trap-PDUs received. [Implementation] Same as the standard. Fixed value of 0.	Y
20	snmpOutTooBigs {snmp 20}	R/O	[Standard] The total number of sent PDUs that have an error status of tooBig. [Implementation] Same as the standard.	Y
21	snmpOutNoSuchNames {snmp 21}	R/O	[Standard] The total number of sent PDUs that have an error status of noSuchName. [Implementation] Same as the standard. SNMP Version 2 and later are excluded.	Y
22	snmpOutBadValues {snmp 22}	R/O	[Standard] The total number of sent PDUs that have an error status of badValue. [Implementation] Same as the standard. SNMP Version 2 and later are excluded.	Y
23	snmpOutReadOnlys {snmp 23}	R/O	[Standard] The total number of sent PDUs that have an error status of readOnly. [Implementation] Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
24	snmpOutGenErrs {snmp 24}	R/O	[Standard] The total number of sent PDUs that have an error status of genErr. [Implementation] Same as the standard.	Y
25	snmpOutGetRequests {snmp 25}	R/O	[Standard] The total number of GetRequestPDUs sent. [Implementation] Same as the standard. Fixed value of 0.	Y
26	snmpOutGetNexts {snmp 26}	R/O	[Standard] The total number of GetNextRequestPDUs sent. [Implementation] Same as the standard. Fixed value of 0.	Y
27	snmpOutSetRequests {snmp 27}	R/O	[Standard] The total number of SetRequestPDUs sent. [Implementation] Same as the standard. Fixed value of 0.	Y
28	snmpOutGetResponses {snmp 28}	R/O	[Standard] The total number of GetResponsePDUs sent. [Implementation] Same as the standard.	Y
29	snmpOutTraps {snmp 29}	R/O	[Standard] The total number of Trap-PDUs sent. [Implementation] Same as the standard.	Y
30	snmpEnableAuthenTraps {snmp 30}	R/NW	[Standard] Indicates whether an authentication-failure trap can be sent: ● enable (1) ● disable (2) [Implementation] Same as the standard. If the configuration command `snmp-server host` has been set, the value is enable.	Y
31	snmpSilentDrops {snmp 31}	R/O	[Standard] The total number of received SNMP messages that are discarded because the response message exceeded the maximum message size. [Implementation] Not implemented.	N

2.10 rmon group (Remote Network Monitoring MIB)

The rmon group is described in the following document:

- RFC 1757

2.10.1 Ethernet Statistics group

(1) ID

```
rmon OBJECT IDENTIFIER ::= {mib-2 16}

statistics OBJECT IDENTIFIER ::= {rmon 1}
Object ID value: 1.3.6.1.2.1.16.1

etherStatsTable OBJECT IDENTIFIER ::= {statistics 1}
Object ID value: 1.3.6.1.2.1.16.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the Ethernet Statistics group.

Table 2-10 Ethernet Statistics group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	etherStatsTable {statistics 1}	NA	[Standard] A table of statistics for an Ethernet interface. [Implementation] Same as the standard.	Y
2	etherStatsEntry {etherStatsTable 1}	NA	[Standard] An entry that contains statistics for a specific Ethernet interface. INDEX {etherStatsIndex} [Implementation] Same as the standard.	Y
3	etherStatsIndex {etherStatsEntry 1}	R/O	[Standard] An Index value that indicates a specific etherStats entry. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
4	etherStatsDataSource {etherStatsEntry 2}	R/NC	[Standard] The object ID of an interface associated with this information. This object instance is the ifIndex of the MIB-II interfaces group. [Implementation] Same as the standard. Read_Only.	Y
5	etherStatsDropEvents {etherStatsEntry 3}	R/O	[Standard] The number of events that a packet has dropped due to insufficient resources. This number indicates the detected packet drops, not the actual packet drops. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	etherStatsOctets {etherStatsEntry 4}	R/O	[Standard] The number of octets (bytes), including bad packets, received over the network. [Implementation] The number of octets (bytes), including bad packets, sent or received over the network. The number of octets is calculated from the MAC header to the FCS field of the frame.	Y
7	etherStatsPkts {etherStatsEntry 5}	R/O	[Standard] The total number of packets received, including bad, broadcast, and multicast packets. [Implementation] The total number of packets sent and received, including bad, broadcast, and multicast packets.	Y
8	etherStatsBroadcastPkts {etherStatsEntry 6}	R/O	[Standard] The number of broadcast packets received, excluding bad and multicast packets. [Implementation] The number of broadcast packets sent and received, excluding bad and multicast packets.	Y
9	etherStatsMulticastPkts {etherStatsEntry 7}	R/O	[Standard] The number of multicast packets received, excluding bad and broadcast packets. [Implementation] The number of multicast packets sent and received, excluding bad and broadcast packets.	Y
10	etherStatsCRCAlignErrors {etherStatsEntry 8}	R/O	[Standard] The number of FCS error packets received. [Implementation] Same as the standard. (Includes received packets containing Symbol Error, in accordance with the definition in sections 24.2.2.1.6 and 22.2.1.5 of IEEE 802.3-2005.)	Y
11	etherStatsUndersizePkts {etherStatsEntry 9}	R/O	[Standard] The number of undersized packets (frame length of less than 64 octets) received. [Implementation] Same as the standard.	Y
12	etherStatsOversizePkts {etherStatsEntry 10}	R/O	[Standard] The number of oversized packets (frame length of more than 1518 octets) received. [Implementation] The number of oversized packets (which exceed the maximum frame length) received.	Y
13	etherStatsFragments {etherStatsEntry 11}	R/O	[Standard] The number of undersized packets (frame length of less than 64 octets) received that have an FCS or Alignment error. [Implementation] Same as the standard.	Y
14	etherStatsJabbers {etherStatsEntry 12}	R/O	[Standard] The number of oversized packets (exceeding the maximum frame length) received that have an FCS or Alignment error. [Implementation] ● gigabitethernet: Fixed value of 0. ● tengigabitethernet: The number of oversized packets (which exceed the maximum frame length) received that have an FCS error	Y

#	Object identifier	Access	Implementation specifications	Implemented
15	etherStatsCollisions {etherStatsEntry 13}	R/O	[Standard] The number of collisions. [Implementation] Same as the standard.	Y
16	etherStatsPkts64Octets {etherStatsEntry 14}	R/O	[Standard] The number of packets received that have a frame length of 64 octets. [Implementation] The number of packets sent and received with a frame length of 64 octets.	Y
17	etherStatsPkts65to127Octets {etherStatsEntry 15}	R/O	[Standard] The number of packets received that have a frame length of from 65 to 127 octets. [Implementation] The number of packets sent and received that have a frame length of from 65 to 127 octets.	Y
18	etherStatsPkts128to255Octets {etherStatsEntry 16}	R/O	[Standard] The number of packets received that have a frame length of from 128 to 255 octets. [Implementation] The number of packets sent and received that have a frame length of from 128 to 255 octets.	Y
19	etherStatsPkts256to511Octets {etherStatsEntry 17}	R/O	[Standard] The number of packets received that have a frame length of from 256 to 511 octets. [Implementation] The number of packets sent and received that have a frame length of from 256 to 511 octets.	Y
20	etherStatsPkts512to1023Octets {etherStatsEntry 18}	R/O	[Standard] The number of packets received with a frame length of from 512 to 1023 octets. [Implementation] The number of packets sent and received that have a frame length of from 512 to 1023 octets.	Y
21	etherStatsPkts1024to1518Octets {etherStatsEntry 19}	R/O	[Standard] The number of packets received that have a frame length of from 1024 to 1518 octets. [Implementation] The number of packets sent and received, with a frame length of 1024 to 1518 octets.	Y
22	etherStatsOwner {etherStatsEntry 20}	R/NC	[Standard] The real entity that makes up this entry and the owner who assigns the resource. [Implementation] Returns a response of the character string `system`. `Read_Only`.	Y
23	etherStatsStatus {etherStatsEntry 21}	R/NC	[Standard] The entry status: ● valid (1) ● createRequest (2) ● underCreation (3) ● invalid (4) [Implementation] Fixed value of valid (1). `Read_Only`.	Y

Note:

The frame length indicates the length from the MAC header to the FCS field. For details about the frame format, see *14.1.3 Control on the MAC and LLC sublayers* in the manual *Configuration Guide Vol.1*.

2.10.2 History Control group

(1) ID

```
rmon OBJECT IDENTIFIER ::= {mib-2 16}

history OBJECT IDENTIFIER ::= {rmon 2}
Object ID value: 1.3.6.1.2.1.16.2

historyControlTable OBJECT IDENTIFIER ::= {history 1}
Object ID value: 1.3.6.1.2.1.16.2.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the History Control group.

Table 2-11 History Control group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	historyControlTable {history 1}	NA	[Standard] Ethernet statistics history control tables. [Implementation] Same as the standard.	Y
2	historyControlEntry {historyControlTable 1}	NA	[Standard] A list of Ethernet statistics history control tables. INDEX {historyControlIndex} [Implementation] Same as the standard. The list can contain no more than 32 entries.	Y
3	historyControlIndex {historyControlEntry 1}	R/O	[Standard] An Index value that indicates a specific historyControl entry. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
4	historyControlDataSource {historyControlEntry 2}#	R/C	[Standard] The object ID of an interface associated with this information. This object instance is the ifIndex of the MIB-II interfaces group. [Implementation] Same as the standard.	Y
5	historyControlBucketsRequested {historyControlEntry 3}#	R/C	[Standard] The number of requested data buckets to be stored in etherHistoryTable.The default is 50. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
6	historyControlBucketsGranted {historyControlEntry 4}	R/O	[Standard] The number of granted data buckets to be stored in etherHistoryTable. The value ranges from 1 to 65535. [Implementation] This number is equal to the value of historyControlBucketsRequested. If the historyControlBucketsRequested value is 50 or more, this number is fixed at 50.	Y
7	historyControlInterval {historyControlEntry 5}#	R/C	[Standard] The sampling interval (in seconds) for the data to be stored in etherHistoryTable. The value ranges from 1 to 3600. The default is 1800. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
8	historyControlOwner {historyControlEntry 6}#	R/C	[Standard] The real entity that makes up this entry and the owner who assigns the resource. [Implementation] A character string of no more than 24 characters can be read and written.	Y
9	historyControlStatus {historyControlEntry 7}	R/C	[Standard] The status of this entry: ● valid (1) ● createRequest (2) ● underCreation (3) ● invalid (4) [Implementation] To add this entry, set createRequest (2) first. Set the MIB in this entry, and then set valid (1). To remove this entry, set invalid (4) first. Next, set createRequest (2), and then acquire this entry. A response of underCreation (3) is then returned. Next, set valid (1), and then acquire this entry. A response of valid (1) is then returned. ● valid (1): Statistics for the interface set in historyControlDataSource can be collected and sampled during the interval set in historyControlInterval. ● invalid (4): Statistics for the interface cannot be acquired. Sampling is also not possible during historyInterval.	Y

#:

This can also be set by using the `rmon collection history` configuration command. If specified in a string, the range of the character codes available is the same as that of the console configuration. If this is changed from the console during setup (while in the state of underCreation) in SNMP manager, all entries that are in the state of underCreation will be deleted.

2.10.3 Ethernet History group

(1) ID

```
rmon OBJECT IDENTIFIER ::= {mib-2 16}

history OBJECT IDENTIFIER ::= {rmon 2}
Object ID value: 1.3.6.1.2.1.16.2

etherHistoryTable OBJECT IDENTIFIER ::= {history 2}
Object ID value: 1.3.6.1.2.1.16.2.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the Ethernet History group.

Table 2-12 Ethernet History group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	etherHistoryTable {history 2}	NA	[Standard] Ethernet statistics history tables. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
2	etherHistoryEntry {etherHistoryTable 1}	NA	[Standard] A list of Ethernet statistics history tables. INDEX {etherHistoryIndex, etherHistorySampleIndex} [Implementation] Same as the standard.	Y
3	etherHistoryIndex {etherHistoryEntry 1}	R/O	[Standard] The same value as the historyControlIndex index value. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
4	etherHistorySampleIndex {etherHistoryEntry 2}	R/O	[Standard] A unique sequence value from the same etherHistoryIndex entry is set, starting at 1. The value ranges from 1 to 2147483647. [Implementation] Same as the standard.	Y
5	etherHistoryIntervalStart {etherHistoryEntry 3}	R/O	[Standard] The time at which the statistics collection was started (in units of 1/100 seconds). [Implementation] Same as the standard.	Y
6	etherHistoryDropEvents {etherHistoryEntry 4}	R/O	[Standard] The number of packet drops that were detected during sampling. [Implementation] Same as the standard.	Y
7	etherHistoryOctets {etherHistoryEntry 5}	R/O	[Standard] The number of octets (bytes) received within a specific period of time. This number includes bad packets. [Implementation] The number of octets (bytes) sent and received within a specific period of time. The number of octets is calculated from the MAC header to the FCS field of the frame.	Y
8	etherHistoryPkts {etherHistoryEntry 6}	R/O	[Standard] The total number of packets received within a specific period of time. This number includes bad, broadcast, and multicast packets. [Implementation] The total number of packets sent and received within a specific period of time.	Y
9	etherHistoryBroadcastPkts {etherHistoryEntry 7}	R/O	[Standard] The number of broadcast packets received within a specific period of time. This number does not include bad or multicast packets. Implementation] The number of broadcast packets sent and received within a specific period of time.	Y
10	etherHistoryMulticastPkts {etherHistoryEntry 8}	R/O	[Standard] The number of multicast packets received within a specific period of time. This number does not include bad or broadcast packets. [Implementation] The number of multicast packets sent and received within a specific period of time.	Y
11	etherHistoryCRCAlignErrors {etherHistoryEntry 9}	R/O	[Standard] The number of FCS error packets received within a specific period of time. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
12	etherHistoryUndersizePkts {etherHistoryEntry 10}	R/O	[Standard] The number of undersized packets (frame length of less than 64 octets) received within a specific period of time. [Implementation] Same as the standard.	Y
13	etherHistoryOversizePkts {etherHistoryEntry 11}	R/O	[Standard] The number of oversized packets (frame length of more than 1518 octets) received within a specific period of time [Implementation] The number of oversized packets (exceeding the maximum frame length) received within a specific period of time.	Y
14	etherHistoryFragments {etherHistoryEntry 12}	R/O	[Standard] The number of undersized packets (frame length of less than 64 octets) received within a specific period of time that have an FCS or Alignment error. [Implementation] Same as the standard.	Y
15	etherHistoryJabbers {etherHistoryEntry 13}	R/O	[Standard] The number of oversized packets (exceeding the maximum frame length) received within a specific period of time that have an FCS or Alignment error. [Implementation] ● gigabitethernet: Fixed value of 0. ● tengigabitethernet: The number of oversized packets (which exceed the maximum frame length) that were received within a specific period of time have an FCS error.	Y
16	etherHistoryCollisions {etherHistoryEntry 14}	R/O	[Standard] The number of collisions that occur within a specific period of time. [Implementation] Same as the standard.	Y
17	etherHistoryUtilization {etherHistoryEntry 15}	R/O	[Standard] The estimated usage rate of the physical layer. The value ranges from 0 to 10000. [Implementation] Indicates the usage rate. The usage rate of a half-duplex line is estimated by: (*number-of-packets* X 160 + *number-of-octets* X 8) / (*time-interval* X *line-speed*) X 10000 The usage rate of a full-duplex line is estimated by: (*number-of-packets* X 160 + *number-of-octets* X 8) / (*time-interval* X *line-speed* X 2) X 10000	Y

2.10.4 Alarm group

(1) ID

```
rmon OBJECT IDENTIFIER ::= {mib-2 16}

alarm OBJECT IDENTIFIER ::= {rmon 3}
Object ID value: 1.3.6.1.2.1.16.3

alarmTable OBJECT IDENTIFIER ::= {alarm 1}
Object ID value: 1.3.6.1.2.1.16.3.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the Alarm group.

Table 2-13 Alarm group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	alarmTable {alarm 1}	NA	[Standard] Alarm tables. [Implementation] Same as the standard.	Y
2	alarmEntry {alarmTable 1}	NA	[Standard] A list of alarm tables. INDEX { alarmIndex } [Implementation] Same as the standard. The list can contain no more than 128 entries.	Y
3	alarmIndex {alarmEntry 1}	R/O	[Standard] An ID that uniquely identifies a row entry in alarmTable. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
4	alarmInterval {alarmEntry 2}#1	R/C	[Standard] The interval (in seconds) between threshold comparisons. The interval ranges from 1 to $2^{32}-1$. [Implementation] Same as the standard.#2	Y
5	alarmVariable {alarmEntry 3}#1	R/C	[Standard] The object identifier of the MIB to be sampled. [Implementation] Same as the standard.	Y
6	alarmSampleType {alarmEntry 4}#1	R/C	[Standard] Indicates the method for performing value and threshold comparison: ● absoluteValue (1) ● deltaValue (2) [Implementation] Same as the standard.	Y
7	alarmValue {alarmEntry 5}	R/O	[Standard] A statistic from the previous sampling. [Implementation] Same as the standard.	Y
8	alarmStartupAlarm {alarmEntry 6}#1	R/C	[Standard] The timing for generating the first alarm: ● risingAlarm (1) ● fallingAlarm (2) ● rising Or fallingAlarm (3) [Implementation] Same as the standard.	Y
9	alarmRisingThreshold {alarmEntry 7}#1	R/C	[Standard] The upper threshold for the sampled statistic. [Implementation] Same as the standard.#2	Y
10	alarmFallingThreshold {alarmEntry 8}#1	R/C	[Standard] The lower threshold for the sampled statistic. [Implementation] Same as the standard.#2	Y
11	alarmRisingEventIndex {alarmEntry 9}#1	R/C	[Standard] The index number of the event group to be used when the upper threshold is exceeded. The value ranges from 0 to 65535. [Implementation] Same as the standard.	Y
12	alarmFallingEventIndex {alarmEntry 10}#1	R/C	[Standard] The index number of the event group to be used when the lower threshold is exceeded. The value ranges from 0 to 65535. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
13	alarmOwner {alarmEntry 11}#1	R/C	[Standard] The real entity that makes up this entry and the owner who assigns the resource. [Implementation] Character strings of no more than 24 characters can be read and written.	Y
14	alarmStatus {alarmEntry 12}	R/C	[Standard] The status of this entry. [Implementation] To add this entry, set createRequest (2) first. Set the MIB in this entry, and then set valid (1). To remove this entry, set invalid (4) first. Next, set createRequest (2), and then acquire this entry. A response of underCreation (3) is then returned. Next, set valid (1), and then acquire this entry. A response of valid (1) is then returned. ● valid (1): Information about the object set in alarmVariable can be sampled during the interval set in alarmInterval. ● invalid (4): The object set in alarmVariable does not exist. Alternatively, the sampling attempted during the interval set in alarmInterval failed.	Y

#1:

This can also be set by using the `rmon alarm` configuration command. For details about the configuration range, see *rmon alarm* in the manual *Configuration Command Reference*. If specified in a string, the range of the character codes available is the same as that of the console configuration. If this is changed from the console during setup (while in the state of underCreation) in SNMP manager, all entries that are in the state of underCreation will be deleted.

#2:

If `2147483648` is set during configuration, `-2147483648` is shown, and the value increments by one. If `4294967295` is set, `-1` is displayed.

2.10.5 Event group

(1) ID

```
rmon OBJECT IDENTIFIER ::= {mib-2 16}

event OBJECT IDENTIFIER ::= {rmon 9}
Object ID value: 1.3.6.1.2.1.16.9

eventTable OBJECT IDENTIFIER ::= {event 1}
Object ID value: 1.3.6.1.2.1.16.9.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the Event group.

Table 2-14 Event group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	eventTable {event 1}	NA	[Standard] A table of events generated by RMON agents. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
2	eventEntry {eventTable 1}	NA	[Standard] A list of events generated by RMON agents. INDEX {eventIndex} [Implementation] Same as the standard. The list can contain no more than 16 entries.	Y
3	eventIndex {eventEntry 1}	R/O	[Standard] The index value of the eventEntry list. This is equivalent to the logEventIndex value in the logEntry list. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
4	eventDescription {eventEntry 2}#	R/C	[Standard] A description of this list. A character string of no more than 127 characters. [Implementation] A character string of no more than 79 characters.	Y
5	eventType {eventEntry 3}#	R/C	[Standard] The event notification method: ● none (1) ● log (2) ● snmp-trap (3) ● log-and-trap (4) [Implementation] Same as the standard.	Y
6	eventCommunity {eventEntry 4}#	R/C	[Standard] The community name of a trap issued when Trap is specified as eventType. A character string of no more than 127 characters. [Implementation] A character string of no more than 60 characters.	Y
7	eventLastTimeSent {eventEntry 5}	R/O	[Standard] The sysUpTime value for when the last event was generated (in units of 1/100 seconds). [Implementation] Same as the standard.	Y
8	eventOwner {eventEntry 6}#	R/C	[Standard] The real entity that makes up this entry and the owner who assigns a resource. No more than 127 characters. [Implementation] Character strings of no more than 24 characters can be read and written.	Y
9	eventStatus {eventEntry 7}	R/C	[Standard] The status of this entry: ● valid (1) ● createRequest (2) ● underCreation (3) ● invalid (4) [Implementation] To add this entry, set createRequest (2) first. Set the MIB in this entry, and then set valid (1). To remove this entry, set invalid (4) first. Next, set createRequest (2), and then acquire this entry. A response of underCreation (3) is then returned. Next, set valid (1), and then acquire this entry. A response of valid (1) is then returned.	Y
10	logTable {event 2}	NA	[Standard] A table of logged events. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
11	logEntry {logTable 1}	NA	[Standard] A list of logged events. INDEX {logEventIndex,logIndex } [Implementation] Same as the standard. The list can contain no more than 128 entries.	Y
12	logEventIndex {logEntry 1}	R/O	[Standard] The index of an event that triggered the generation of this log. This value indicates the event that has the same eventIndex value. The value ranges from 1 to 65535. [Implementation] Same as the standard.	Y
13	logIndex {logEntry 2}	R/O	[Standard] The index of the log for the same event. The value ranges from 1 to 2147483647. [Implementation] Same as the standard.	Y
14	logTime {logEntry 3}	R/O	[Standard] The sysUpTime value for when this log list was generated. [Implementation] Same as the standard.	Y
15	logDescription {logEntry 4}	R/O	[Standard] A comment about the source event for this log list. A character string of no more than 255 characters. [Implementation] Returns a response containing a character string of no more than 72 characters.	Y

#:

This can also be set by using the `rmon event` configuration command. If specified in a string, the range of the character codes available is the same as that of the console configuration. If this is changed from the console during setup (while in the state of underCreation) in SNMP manager, all entries that are in the state of underCreation will be deleted.

2.11 dot1dBridge group

The dot1dBridge group is described in the following documents:

- RFC 1493
- RFC 2674

2.11.1 dot1dBase group

(1) ID

```
dot1dBridge OBJECT IDENTIFIER ::= {mib-2 17}

dot1dBase OBJECT IDENTIFIER ::= {dot1dBridge 1}
Object ID value: 1.3.6.1.2.1.17.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1dBase group.

Table 2-15 dot1dBase group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1dBaseBridgeAddress {dot1dBase 1}	R/O	[Standard] The MAC address of a bridge. [Implementation] Same as the standard.	Y
2	dot1dBaseNumPorts {dot1dBase 2}	R/O	[Standard] The number of ports on the bridge. [Implementation] Same as the standard.	Y
3	dot1dBaseType {dot1dBase 3}	R/O	[Standard] The bridging type supported by the bridge: ● unknown (1) ● transparent-only (2) ● sourceroute-only (3) ● srt (4) [Implementation] Fixed value of transparent-only (2).	Y
4	dot1dBasePortTable {dot1dBase 4}	NA	[Standard] A table of information about each bridge port. [Implementation] Same as the standard.	Y
5	dot1dBasePortEntry {dot1dBasePortTable 1}	NA	[Standard] A list of information about each bridge port. INDEX {dot1dBasePort} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	dot1dBasePort {dot1dBasePortEntry 1}	R/O	[Standard] The port number of a port (1-65535). ● Port number of a physical port Any physical ports are uniquely identified. Port number: Physical port number ● Port number of a channel group Calculated from the channel group number in link aggregation as follows: Port number: 65 (fixed value) + channel group number ● Peer link Port number: 130 (fixed value) [Implementation] Same as the standard.	Y
7	dot1dBasePortIfIndex {dot1dBasePortEntry 2}	R/O	[Standard] The interface that corresponds to this port is the instance value of the object defined in MIB-II. [Implementation] Same as the standard.	Y
8	dot1dBasePortCircuit {dot1dBasePortEntry 3}	R/O	[Standard] The identifier of the port with the same instance value set in the dot1dBasePortIfIndex. [Implementation] Fixed value of {0.0}.	Y
9	dot1dBasePortDelayExceededDiscards {dot1dBasePortEntry 4}	R/O	[Standard] The total number of frames discarded due to a pass-through delay. [Implementation] Fixed value of 0.	M
10	dot1dBasePortMtuExceededDiscards {dot1dBasePortEntry 5}	R/O	[Standard] The total number of frames discarded due to a data overflow. [Implementation] Fixed value of 0.	M

2.11.2 dot1dStp group

(1) ID

```
dot1dBridge OBJECT IDENTIFIER ::= {mib-2 17}

dot1dStp OBJECT IDENTIFIER ::= {dot1dBridge 2}
Object ID value: 1.3.6.1.2.1.17.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1dStp group.

The dot1dStp group is enabled only when it operates in Single Spanning Tree mode. For PVST+ in a Rapid Spanning Tree Protocol, Multiple Spanning Tree, or without Single Spanning Tree, dot1dStp group objects will be treated as dummy values.

In addition, dot1dStpPortTable includes only ports on which Single Spanning Tree is running.

Table 2-16 dot1dStp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1dStpProtocolSpecification {dot1dStp 1}	R/O	[Standard] The version of the Spanning Tree Protocol run by the bridge: ● unknown (1) ● decLb100 (2) ● ieee8021d (3) [Implementation] Fixed value of 3.	Y
2	dot1dStpPriority {dot1dStp 2}	R/NW	[Standard] The priority value of the bridge (0-65535). [Implementation] Same as the standard.	Y
3	dot1dStpTimeSinceTopologyChange {dot1dStp 3}	R/O	[Standard] The time (in units of 1/100 seconds) that has elapsed since a topology change occurred. [Implementation] Same as the standard.	Y
4	dot1dStpTopChanges {dot1dStp 4}	R/O	[Standard] The number of topology changes. [Implementation] Same as the standard.	Y
5	dot1dStpDesignatedRoot {dot1dStp 5}	R/O	[Standard] A root bridge identification value held by the bridge. [Implementation] Same as the standard.	Y
6	dot1dStpRootCost {dot1dStp 6}	R/O	[Standard] A root path cost value held by the bridge. [Implementation] Same as the standard.	Y
7	dot1dStpRootPort {dot1dStp 7}	R/O	[Standard] A root port value held by the bridge. [Implementation] Same as the standard. A value of 0 indicates that no root port exists. The ifIndex value of the VLAN used for the link is shown as a virtual link.	Y
8	dot1dStpMaxAge {dot1dStp 8}	R/O	[Standard] The maximum aging time (in units of 1/100 seconds) held by the bridge. [Implementation] Same as the standard.	Y
9	dot1dStpHelloTime {dot1dStp 9}	R/O	[Standard] The Hello time (in units of 1/100 seconds) held by the bridge. [Implementation] Same as the standard.	Y
10	dot1dStpHoldTime {dot1dStp 10}	R/O	[Standard] The Hold time (in units of 1/100 seconds) held by the bridge. [Implementation] Same as the standard.	Y
11	dot1dStpForwardDelay {dot1dStp 11}	R/O	[Standard] The transfer delay time (in units of 1/100 seconds) held by the bridge. [Implementation] Same as the standard.	Y
12	dot1dStpBridgeMaxAge {dot1dStp 12}	R/NW	[Standard] The maximum aging time (600-4000 in units of 1/100 seconds) set in the bridge. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
13	dot1dStpBridgeHelloTime {dot1dStp 13}	R/NW	[Standard] The Hello time (100-1000 in units of 1/100 seconds) set in the bridge. [Implementation] Same as the standard.	Y
14	dot1dStpBridgeForwardDelay {dot1dStp 14}	R/NW	[Standard] The transfer delay time (400-3000 in units of 1/100 seconds) set in the bridge. [Implementation] Same as the standard.	Y
15	dot1dStpPortTable {dot1dStp 15}	NA	[Standard] A table of port information for the Spanning Tree Protocol. [Implementation] Same as the standard.	Y
16	dot1dStpPortEntry {dot1dStpPortTable 1}	NA	[Standard] A list of Spanning Tree Protocol status information for each port INDEX { dot1dStpPort } [Implementation] Same as the standard.	Y
17	dot1dStpPort {dot1dStpPortEntry 1}	R/O	[Standard] The port number of a port under the Spanning Tree Protocol (1-65535). [Implementation] Same as the standard.	Y
18	dot1dStpPortPriority {dot1dStpPortEntry 2}	R/NW	[Standard] The priority of the port (0-255). [Implementation] Same as the standard.	Y
19	dot1dStpPortState {dot1dStpPortEntry 3}	R/O	[Standard] The current status of the port: ● disabled (1) ● blocking (2) ● listening (3) ● learning (4) ● forwarding (5) ● broken (6) [Implementation] For linkdown, the value is disabled (1); broken (6) is not used.	Y
20	dot1dStpPortEnable {dot1dStpPortEntry 4}	R/NW	[Standard] The enabled or disabled status of the port: ● enabled (1) ● disabled (2) [Implementation] Same as the standard.	Y
21	dot1dStpPortPathCost {dot1dStpPortEntry 5}	R/NW	[Standard] The path cost value for the port (1-65535). [Implementation] 0-200000000. A value of 0 indicates that the port link is down.	Y
22	dot1dStpPortDesignatedRoot {dot1dStpPortEntry 6}	R/O	[Standard] The root bridge identifier in the configured BPDU. [Implementation] Same as the standard.	Y
23	dot1dStpPortDesignatedCost {dot1dStpPortEntry 7}	R/O	[Standard] The path cost value for the specified port. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
24	dot1dStpPortDesignatedBridge {dot1dStpPortEntry 8}	R/O	[Standard] The bridge identifier for the specified bridge. [Implementation] Same as the standard.	Y
25	dot1dStpPortDesignatedPort {dot1dStpPortEntry 9}	R/O	[Standard] The port identifier for the specified bridge: ● SIZE (2) [Implementation] Same as the standard.	Y
26	dot1dStpPortForwardTransitions {dot1dStpPortEntry 10}	R/O	[Standard] The number of times the port state has changed from learning to transferring. [Implementation] Same as the standard.	Y

2.11.3 dot1dTp group

(1) ID

```
dot1dBridge OBJECT IDENTIFIER ::= {mib-2 17}

dot1dTp OBJECT IDENTIFIER ::= {dot1dBridge 4}
Object ID value: 1.3.6.1.2.1.17.4
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1dTp group.

Table 2-17 dot1dTp group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1dTpLearnedEntryDiscards {dot1dTp 1}	R/O	[Standard] The number of forwarding entries discarded due to insufficient space in the forwarding database. [Implementation] Fixed value of 0.	M
2	dot1dTpAgingTime {dot1dTp 2}	R/NW	[Standard] The timeout time (10-1000000 seconds) before aging out a forwarding entry that was dynamically learned. [Implementation] Same as the standard. A value of 0 indicates that aging was disabled during configuration.	Y
3	dot1dTpFdbTable {dot1dTp 3}	NA	[Standard] A table of unicast entries that contain filtering information. [Implementation] Synthesizes the MAC address tables of all VLANs. If more than one VLAN has the same MAC address, the address of the VLAN with the lowest number takes priority.	Y
4	dot1dTpFdbEntry {dot1dTpFdbTable 1}	NA	[Standard] A unicast MAC address that contains filtering information. INDEX {dot1dTpFdbAddress} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
5	dot1dTpFdbAddress {dot1dTpFdbEntry 1}	R/O	[Standard] A unicast MAC address that contains filtering information. [Implementation] Same as the standard.	Y
6	dot1dTpFdbPort {dot1dTpFdbEntry 2}	R/O	[Standard] The port number of the port that sent a frame that has the same source address value as the corresponding instance value of dot1dTpFdbAddress. [Implementation] Same as the standard. This configuration is used for static entries.	Y
7	dot1dTpFdbStatus {dot1dTpFdbEntry 3}	R/O	[Standard] The status of the MAC address table: ● other (1) ● invalid (2) ● learned (3) ● self (4) ● mgmt (5) [Implementation] Same as the standard.	Y
8	dot1dTpPortTable {dot1dTp 4}	NA	[Standard] A table of information about all ports. [Implementation] Same as the standard.	Y
9	dot1dTpPortEntry {dot1dTpPortTable 1}	NA	[Standard] A list of information about each port. INDEX {dot1dTpPort} [Implementation] Same as the standard.	Y
10	dot1dTpPort {dot1dTpPortEntry 1}	R/O	[Standard] A port number (1-65535) that indicates the port associated with the management information in this entry. [Implementation] Same as the standard.	Y
11	dot1dTpPortMaxInfo {dot1dTpPortEntry 2}	R/O	[Standard] The maximum size of a send-and-receive information field for this port. [Implementation] Same as the standard.	Y
12	dot1dTpPortInFrames {dot1dTpPortEntry 3}	R/O	[Standard] The number of frames received by this port. [Implementation] Same as the standard.	Y
13	dot1dTpPortOutFrames {dot1dTpPortEntry 4}	R/O	[Standard] The number of frames sent by this port. [Implementation] Same as the standard.	Y
14	dot1dTpPortInDiscards {dot1dTpPortEntry 5}	R/O	[Standard] The number of valid frames received but discarded. [Implementation] Same as the standard.	Y
15	dot1dTpHCPortTable {dot1dTp 5}	NA	[Standard] A table of information about a high-capacity port. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
16	dot1dTpHCPortEntry {dot1dTpHCPortTable 1}	NA	[Standard] A list of information about a high-capacity port. INDEX {dot1dTpPort} [Implementation] Same as the standard.	Y
17	dot1dTpHCPortInFrames {dot1dTpHCPortEntry 1}	R/O	[Standard] The number of frames received by the high-capacity port. [Implementation] Same as the standard.	Y
18	dot1dTpHCPortOutFrames {dot1dTpHCPortEntry 2}	R/O	[Standard] The number of frames sent by the high-capacity port. [Implementation] Same as the standard.	Y
19	dot1dTpHCPortInDiscards {dot1dTpHCPortEntry 3}	R/O	[Standard] The number of frames received but discarded by the high-capacity port. [Implementation] Same as the standard.	Y
20	dot1dTpPortOverflowTable {dot1dTp 6}	NA	[Standard] A table of overflow information for the high-capacity port. [Implementation] Same as the standard.	Y
21	dot1dTpPortOverflowEntry {dot1dTpPortOverflowTable 1}	NA	[Standard] A list of overflow information for the high-capacity port. INDEX {dot1dTpPort} [Implementation] Same as the standard.	Y
22	dot1dTpPortInOverflowFrames {dot1dTpPortOverflowEntry 1}	R/O	[Standard] The number of times the dot1dTpPortInFrames counter overflowed. [Implementation] Fixed value of 0.	M
23	dot1dTpPortOutOverflowFrames {dot1dTpPortOverflowEntry 2}	R/O	[Standard] The number of times the dot1dTpPortOutFrames counter overflowed. [Implementation] Fixed value of 0.	M
24	dot1dTpPortInOverflowDiscards {dot1dTpPortOverflowEntry 3}	R/O	[Standard] The number of times the dot1dTpPortInDiscards counter has overflowed. [Implementation] Fixed value of 0.	M

2.11.4 pBridgeMIB group

(1) ID

```
dot1dBridge OBJECT IDENTIFIER ::= {mib-2 17}

pBridgeMIB OBJECT IDENTIFIER ::= {dot1dBridge 6}
Object ID value: 1.3.6.1.2.1.17.6

pBridgeMIBObjects  OBJECT IDENTIFIER ::= {pBridgeMIB 1}
dot1dExtBase       OBJECT IDENTIFIER ::= {pBridgeMIBObjects 1}
dot1dPriority      OBJECT IDENTIFIER ::= {pBridgeMIBObjects 2}
dot1dGarp          OBJECT IDENTIFIER ::= {pBridgeMIBObjects 3}
dot1dGmrp          OBJECT IDENTIFIER ::= {pBridgeMIBObjects 4}
```

```
pBridgeConformance OBJECT IDENTIFIER ::= {pBridgeMIB 2}
pBridgeGroups      OBJECT IDENTIFIER ::= {pBridgeConformance 1}
pBridgeCompliances OBJECT IDENTIFIER ::= {pBridgeConformance 2}
```

(2) Implementation specifications

The following table shows the implementation specifications for the pBridgeMIB group.

Table 2-18 pBridgeMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1dDeviceCapabilities {dot1dExtBase 1}	R/O	[Standard] IEEE 802.1D or 802.1Q options implemented in the device: ● dot1dExtendedFilteringServices (0) ● dot1dTrafficClasses (1) ● dot1qStaticEntryIndividualPort (2) ● dot1qIVLCapable (3) ● dot1qSVLCapable (4) ● dot1qHybridCapable (5) ● dot1qConfigurablePvidTagging (6) ● dot1dLocalVlanCapable (7) [Implementation] {dot1dTrafficClasses (1), dot1qIVLCapable (3), dot1qConfigurablePvidTagging (6)} Manager displays the value as text.	Y
2	dot1dTrafficClassesEnabled {dot1dExtBase 2}	R/NW	[Standard] The traffic class support status of the bridge: ● true (1) ● false (2) [Implementation] true (1)	Y
3	dot1dGmrpStatus {dot1dExtBase 3}	R/NW	[Standard] The GMRP status: ● enabled (1) ● disabled (2) [Implementation] disabled (2)	Y
4	dot1dPortCapabilitiesTable {dot1dExtBase 4}	NA	[Standard] A table of port capacity information. [Implementation] Same as the standard.	Y
5	dot1dPortCapabilitiesEntry {dot1dPortCapabilitiesTable 1}	NA	[Standard] A list of port capacity information. [Implementation] Same as the standard.	Y
6	dot1dPortCapabilities {dot1dPortCapabilitiesEntry 1}	R/O	[Standard] The IEEE 802.1D and 802.1Q status of the port: ● dot1qDot1qTagging (0) ● dot1qConfigurableAcceptableFrameTypes (1) ● dot1qIngressFiltering (2) [Implementation] dot1qIngressFiltering (2) Manager displays the value as text.	Y
7	dot1dPortPriorityTable {dot1dPriority 1}	NA	[Standard] A table of port priority information. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
8	dot1dPortPriorityEntry {dot1dPortPriorityTable 1}	NA	[Standard] A list of port priority information. [Implementation] Same as the standard.	Y
9	dot1dPortDefaultUserPriority {dot1dPortPriorityEntry 1}	R/NW	[Standard] The default ingress user priority for ports (0-7). [Implementation] 0	Y
10	dot1dPortNumTrafficClasses {dot1dPortPriorityEntry 2}	R/NW	[Standard] The ingress traffic class number for ports (1-8). [Implementation] 1	Y
11	dot1dTrafficClassTable {dot1dPriority 3}	NA	[Standard] A table of traffic class information. [Implementation] Same as the standard.	Y
12	dot1dTrafficClassEntry {dot1dTrafficClassTable 1}	NA	[Standard] A list of traffic class information. INDEX {dot1dBasePort, dot1dTrafficClassPriority} [Implementation] Same as the standard.	Y
13	dot1dTrafficClassPriority {dot1dTrafficClassEntry 1}	NA	[Standard] The priority of the traffic class (0-7). [Implementation] Same as the standard.	Y
14	dot1dTrafficClass {dot1dTrafficClassEntry 2}	R/NW	[Standard] The traffic class (0-7). [Implementation] Same as the standard.	Y

2.11.5 qBridgeMIB group

(1) ID

```
dot1dBridge OBJECT IDENTIFIER ::= {mib-2 17}

qBridgeMIB OBJECT IDENTIFIER ::= {dot1dBridge 7}
Object ID value: 1.3.6.1.2.1.17.7

qBridgeMIBObjects  OBJECT IDENTIFIER ::= {qBridgeMIB 1}
dot1qBase          OBJECT IDENTIFIER ::= {qBridgeMIBObjects 1}
dot1qTp            OBJECT IDENTIFIER ::= {qBridgeMIBObjects 2}
dot1qStatic        OBJECT IDENTIFIER ::= {qBridgeMIBObjects 3}
dot1qVlan          OBJECT IDENTIFIER ::= {qBridgeMIBObjects 4}
qBridgeConformance OBJECT IDENTIFIER ::= {qBridgeMIB 2}
qBridgeGroups      OBJECT IDENTIFIER ::= {qBridgeConformance 1}
qBridgeCompliances OBJECT IDENTIFIER ::= {qBridgeConformance 2}
dot1dPortPair      OBJECT IDENTIFIER ::= {dot1dBridge 10}
```

(2) Implementation specifications

The following table shows the implementation specifications for the qBridgeMIB group.

Table 2-19 qBridgeMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1qVlanVersion Number {dot1qBase 1}	R/O	[Standard] The version number of IEEE 802.1Q. ● version1 (1) [Implementation] 1	Y
2	dot1qMaxVlanId {dot1qBase 2}	R/O	[Standard] The maximum number of IEEE 802.1Q VLAN IDs. [Implementation] 4094	Y
3	dot1qMaxSupportedVlans {dot1qBase 3}	R/O	[Standard] The maximum number of IEEE 802.1Q VLANs. [Implementation] 4094	Y
4	dot1qNumVlans {dot1qBase 4}	R/O	[Standard] The current number of IEEE 802.1Q VLANs. [Implementation] Same as the standard.	Y
5	dot1qGvrpStatus {dot1qBase 5}	R/NW	[Standard] The management status of GVRP. [Implementation] disabled (2)	Y
6	dot1qFdbTable {dot1qTp 1}	NA	[Standard] A table of MAC address tables. [Implementation] Not implemented.	N
7	dot1qFdbEntry {dot1qFdbTable 1}	NA	[Standard] A list of MAC address tables. INDEX {dot1qFdbId } [Implementation] Same as the standard.	Y
8	dot1qFdbId {dot1qFdbEntry 1}	NA	[Standard] The identifier of a MAC address table. [Implementation] Not implemented.	N
9	dot1qFdbDynamicCount {dot1qFdbEntry 2}	R/O	[Standard] The number of dynamic entries in the MAC address table. [Implementation] Not implemented.	N
10	dot1qTpFdbTable {dot1qTp 2}	NA	[Standard] A table of information about the transparent MAC address table. [Implementation] Same as the standard.#	Y
11	dot1qTpFdbEntry {dot1qTpFdbTable 1}	NA	[Standard] A list of information about the transparent MAC address table. INDEX {dot1qFdbId, dot1qTpFdbAddress } [Implementation] Same as the standard.#	Y
12	dot1qTpFdbAddress {dot1qTpFdbEntry 1}	NA	[Standard] A unicast MAC address in the transparent MAC address table. [Implementation] Same as the standard.#	Y
13	dot1qTpFdbPort {dot1qTpFdbEntry 2}	R/O	[Standard] A port number in the transparent MAC address table (0-65535) [Implementation] Same as the standard. This configuration is used for static entries.#	Y

#	Object identifier	Access	Implementation specifications	Implemented
14	dot1qTpFdbStatus {dot1qTpFdbEntry 3}	R/O	[Standard] The status of the MAC address table: ● other (1) ● invalid (2) ● learned (3) ● self (4) ● mgmt (5) [Implementation] Returns learned (3) for a dynamic entry. Returns mgmt (5) for static entries or dotlx entries.#	Y
15	dot1qTpGroupTable {dot1qTp 3}	NA	[Standard] A table of information about a transparent group. [Implementation] Not implemented.	N
16	dot1qTpGroupEntry {dot1qTpGroupTable 1}	NA	[Standard] A list of information about a transparent group. INDEX {dot1qVlanIndex, dot1qTpGroupAddress} [Implementation] Not implemented.	N
17	dot1qTpGroupAddress {dot1qTpGroupEntry 1}	NA	[Standard] A destination MAC address in the transparent group. [Implementation] Not implemented.	N
18	dot1qTpGroupEgressPorts {dot1qTpGroupEntry 2}	R/O	[Standard] The full set of ingress ports in the transparent group. [Implementation] Not implemented.	N
19	dot1qTpGroupLearnt {dot1qTpGroupEntry 3}	R/O	[Standard] A subset of learned ports in the transparent group. [Implementation] Not implemented.	N
20	dot1qForwardAllTable {dot1qTp 4}	NA	[Standard] A table of forwarding information for a VLAN that forwards all multicasts. [Implementation] Not implemented.	N
21	dot1qForwardAllEntry {dot1qForwardAllTable 1}	NA	[Standard] A list of forwarding information for a VLAN that forwards all multicasts. INDEX {dot1qVlanIndex} [Implementation] Not implemented.	N
22	dot1qForwardAllPorts {dot1qForwardAllEntry 1}	R/O	[Standard] The full set of VLAN ports that forward all multicast group addresses. [Implementation] Not implemented.	N
23	dot1qForwardAllStaticPorts {dot1qForwardAllEntry 2}	R/NW	[Standard] A set of static VLAN ports that forward all multicast group addresses. [Implementation] Not implemented.	N
24	dot1qForwardAllForbiddenPorts {dot1qForwardAllEntry 3}	R/NW	[Standard] A set of VLAN ports that do not forward any multicast group addresses. [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
25	dot1qForwardUnregisteredTable {dot1qTp 5}	NA	[Standard] A table of forwarding information for a VLAN that forwards unregistered multicast group addresses. [Implementation] Not implemented.	N
26	dot1qForwardUnregisteredEntry {dot1qForwardUnregisteredTable 1}	NA	[Standard] A list of forwarding information for a VLAN that forwards unregistered multicast group addresses. INDEX {dot1qVlanIndex} [Implementation] Not implemented.	N
27	dot1qForwardUnregisteredPorts {dot1qForwardUnregisteredEntry 1}	R/O	[Standard] The full set of VLAN ports that forward unregistered multicast group addresses. [Implementation] Not implemented.	N
28	dot1qForwardUnregisteredStaticPorts {dot1qForwardUnregisteredEntry 2}	R/NW	[Standard] A set of static VLAN ports that forward unregistered multicast group addresses. [Implementation] Not implemented.	N
29	dot1qForwardUnregisteredForbiddenPorts {dot1qForwardUnregisteredEntry 3}	R/NW	[Standard] A set of static VLAN ports that do not forward any unregistered multicast group addresses. [Implementation] Not implemented.	N
30	dot1qStaticUnicastTable {dot1qStatic 1}	NA	[Standard] A table of filtering information for static unicast MAC addresses. [Implementation] Not implemented.	N
31	dot1qStaticUnicastEntry {dot1qStaticUnicastTable 1}	NA	[Standard] A list of filtering information for static unicast MAC addresses. INDEX {dot1qFdbId, dot1qStaticUnicastAddress, dot1qStaticUnicastReceivePort} [Implementation] Not implemented.	N
32	dot1qStaticUnicastAddress {dot1qStaticUnicastEntry 1}	NA	[Standard] A destination MAC address for static unicast. [Implementation] Not implemented.	N
33	dot1qStaticUnicastReceivePort {dot1qStaticUnicastEntry 2}	NA	[Standard] The port number (0-65535) of a port that receives static unicast addresses. [Implementation] Not implemented.	N
34	dot1qStaticUnicastAllowedToGoTo {dot1qStaticUnicastEntry 3}	R/NW	[Standard] A set of ports that flood static unicast addresses. [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
35	dot1qStaticUnicastStatus {dot1qStaticUnicastEntry 4}	R/NW	[Standard] The entry status of a static unicast address: ● other (1) ● invalid (2) ● permanent (3) ● deleteOnReset (4) ● deleteOnTimeout (5) [Implementation] Not implemented.	N
36	dot1qStaticMulticastTable {dot1qStatic 2}	NA	[Standard] A table of filtering information for a VLAN that forwards static multicast and broadcast MAC addresses. [Implementation] Not implemented.	N
37	dot1qStaticMulticastEntry {dot1qStaticMulticastTable 1}	NA	[Standard] A list of filtering information for a VLAN that forwards static multicast and broadcast MAC addresses. INDEX {dot1qVlanIndex, dot1qStaticMulticastAddress, dot1qStaticMulticastReceivePort} [Implementation] Not implemented.	N
38	dot1qStaticMulticastAddress {dot1qStaticMulticastEntry 1}	NA	[Standard] A destination MAC address for static multicast or broadcast. [Implementation] Not implemented.	N
39	dot1qStaticMulticastReceivePort {dot1qStaticMulticastEntry 2}	NA	[Standard] The port number of a port that receives static multicast or broadcast MAC addresses (0-65535). [Implementation] Not implemented.	N
40	dot1qStaticMulticastStaticEgressPorts {dot1qStaticMulticastEntry 3}	R/NW	[Standard] A set of ports that forward static multicast or broadcast MAC addresses. [Implementation] Not implemented.	N
41	dot1qStaticMulticastForbiddenEgressPorts {dot1qStaticMulticastEntry 4}	R/NW	[Standard] A set of ports that do not forward static multicast or broadcast MAC addresses. [Implementation] Not implemented.	N
42	dot1qStaticMulticastStatus {dot1qStaticMulticastEntry 5}	R/NW	[Standard] The entry status of static multicast or broadcast: ● other (1) ● invalid (2) ● permanent (3) ● deleteOnReset (4) ● deleteOnTimeout (5) [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
43	dot1qVlanNumDeletes {dot1qVlan 1}	R/O	[Standard] The number of VLAN entry deletions. [Implementation] Same as the standard.	Y
44	dot1qVlanCurrentTable {dot1qVlan 2}	NA	[Standard] A table of the current configuration information for a VLAN. [Implementation] Same as the standard.	Y
45	dot1qVlanCurrentEntry {dot1qVlanCurrentTable 1}	NA	[Standard] A list of the current configuration information for a VLAN. INDEX {dot1qVlanTimeMark, dot1qVlanIndex} [Implementation] Same as the standard.	Y
46	dot1qVlanTimeMark {dot1qVlanCurrentEntry 1}	NA	[Standard] A time filter for entries. [Implementation] Same as the standard.	Y
47	dot1qVlanIndex {dot1qVlanCurrentEntry 2}	NA	[Standard] A VLAN ID. [Implementation] Same as the standard.	Y
48	dot1qVlanFdbId {dot1qVlanCurrentEntry 3}	R/O	[Standard] The ID of a MAC address table used by the VLAN. [Implementation] Same as the standard.	Y
49	dot1qVlanCurrentEgressPorts {dot1qVlanCurrentEntry 4}	R/O	[Standard] A set of VLAN ports that send tagged or untagged frame traffic. [Implementation] Same as the standard.	Y
50	dot1qVlanCurrentUntaggedPorts {dot1qVlanCurrentEntry 5}	R/O	[Standard] A set of VLAN ports that send untagged frame traffic. [Implementation] Same as the standard.	Y
51	dot1qVlanStatus {dot1qVlanCurrentEntry 6}	R/O	[Standard] The VLAN status: ● other (1) ● permanent (2) ● dynamicGvrp (3) [Implementation] Same as the standard.	Y
52	dot1qVlanCreationTime {dot1qVlanCurrentEntry 7}	R/O	[Standard] The sysUpTime value for when the VLAN was created. [Implementation] Same as the standard.	Y
53	dot1qVlanStaticTable {dot1qVlan 3}	NA	[Standard] A table of static configuration information for a VLAN. [Implementation] Same as the standard.	Y
54	dot1qVlanStaticEntry {dot1qVlanStaticTable 1}	NA	[Standard] A list of static configuration information for a VLAN. INDEX {dot1qVlanIndex} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
55	dot1qVlanStaticName {dot1qVlanStaticEntry 1}	R/NC	[Standard] The static identification name of the VLAN. [Implementation] Same as the standard.	Y
56	dot1qVlanStaticEgressPorts {dot1qVlanStaticEntry 2}	R/NC	[Standard] A set of ports in the static egress list for the VLAN. [Implementation] Same as the standard.	Y
57	dot1qVlanForbiddenEgressPorts {dot1qVlanStaticEntry 3}	R/NC	[Standard] A set of ports that are forbidden from being registered in the VLAN egress list. [Implementation] Same as the standard.	Y
58	dot1qVlanStaticUntaggedPorts {dot1qVlanStaticEntry 4}	R/NC	[Standard] A set of untagged ports that send VLAN egress packets. [Implementation] Same as the standard.	Y
59	dot1qVlanStaticRowStatus {dot1qVlanStaticEntry 5}	R/NC	[Standard] The entry status. [Implementation] Same as the standard.	Y
60	dot1qNextFreeLocalVlanIndex {dot1qVlan 4}	R/O	[Standard] The next available VLAN index (0, or 4096-2147483647). [Implementation] 0 or 4096.	Y
61	dot1qPortVlanTable {dot1qVlan 5}	NA	[Standard] A table of VLAN configuration information for the port. [Implementation] Same as the standard.	Y
62	dot1qPortVlanEntry {dot1qPortVlanTable 1}	NA	[Standard] A list of VLAN configuration information for the port. [Implementation] Same as the standard.	Y
63	dot1qPvid {dot1qPortVlanEntry 1}	R/NW	[Standard] A PVID VLAN ID assigned to untagged frames or Priority-Tagged frames. [Implementation] Same as the standard.	Y
64	dot1qPortAcceptableFrameTypes {dot1qPortVlanEntry 2}	R/NW	[Standard] Determines the frame type that can be received by the port: ● admitAll (1) ● admitOnlyVlanTagged (2) [Implementation] admitAll (1)	Y
65	dot1qPortIngressFiltering {dot1qPortVlanEntry 3}	R/NW	[Standard] Filters frames to the port. [Implementation] Same as the standard.	Y
66	Dot1qPortGvrpStatus {dot1qPortVlanEntry 4}	R/NW	[Standard] The GVRP status of the port. [Implementation] disabled (2)	Y

#	Object identifier	Access	Implementation specifications	Implemented
67	Dot1qPortGvrpFailedRegistrations {dot1qPortVlanEntry 5}	R/O	[Standard] The total number of port GVRP registrations that failed. [Implementation] Not implemented.	N
68	dot1qPortGvrpLastPduOrigin {dot1qPortVlanEntry 6}	R/O	[Standard] The source MAC address for the last GVRP received by the port. [Implementation] Not implemented.	N
69	dot1qPortVlanStatisticsTable {dot1qVlan 6}	NA	[Standard] A table of VLAN statistics for the port. [Implementation] Same as the standard.	Y
70	dot1qPortVlanStatisticsEntry {dot1qPortVlanStatisticsTable 1}	NA	[Standard] A list of VLAN statistics for the port. INDEX {dot1dBasePort, dot1qVlanIndex} [Implementation] Same as the standard.	Y
71	dot1qTpVlanPortInFrames {dot1qPortVlanStatisticsEntry 1}	R/O	[Standard] The number of valid frames received by the VLAN port. [Implementation] Fixed value of 0.	M
72	dot1qTpVlanPortOutFrames {dot1qPortVlanStatisticsEntry 2}	R/O	[Standard] The number of valid frames sent by the VLAN port. [Implementation] Fixed value of 0.	M
73	dot1qTpVlanPortInDiscards {dot1qPortVlanStatisticsEntry 3}	R/O	[Standard] The number of valid frames received by the VLAN port but discarded. [Implementation] Fixed value of 0.	M
74	dot1qTpVlanPortInOverflowFrames {dot1qPortVlanStatisticsEntry 4}	R/O	[Standard] The number of times the dot1qTpVlanPortInFrames counter overflowed. [Implementation] Fixed value of 0.	M
75	dot1qTpVlanPortOutOverflowFrames {dot1qPortVlanStatisticsEntry 5}	R/O	[Standard] The number of times the dot1qTpVlanPortOutFrames counter overflowed. [Implementation] Fixed value of 0.	M
76	dot1qTpVlanPortInOverflowDiscards {dot1qPortVlanStatisticsEntry 6}	R/O	[Standard] The number of times the dot1qTpVlanPortInDiscards counter overflowed. [Implementation] Fixed value of 0.	M
77	dot1qPortVlanHCStatisticsTable {dot1qVlan 7}	NA	[Standard] A table of VLAN high-capacity statistics for the port. [Implementation] Same as the standard.	Y
78	dot1qPortVlanHCStatisticsEntry {dot1qPortVlanHCStatisticsTable 1}	NA	[Standard] A list of VLAN high-capacity statistics for the port. INDEX {dot1dBasePort, dot1qVlanIndex} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
79	dot1qTpVlanPortHCInFrames {dot1qPortVlanHCStatisticsEntry 1}	R/O	[Standard] The number of valid frames received by the VLAN port. [Implementation] Fixed value of 0.	M
80	dot1qTpVlanPortHCOutFrames {dot1qPortVlanHCStatisticsEntry 2}	R/O	[Standard] The number of valid frames sent by the VLAN port. [Implementation] Fixed value of 0.	M
81	dot1qTpVlanPortHCInDiscards {dot1qPortVlanHCStatisticsEntry 3}	R/O	[Standard] The number of valid frames received by the VLAN port but discarded. [Implementation] Fixed value of 0.	M
82	dot1qLearningConstraintsTable {dot1qVlan 8}	NA	[Standard] A table of learning constraints. [Implementation] This table is always empty, in accordance with the device specifications.	Y
83	dot1qLearningConstraintsEntry {dot1qLearningConstraintsTable 1}	NA	[Standard] A list of learning constraints. INDEX {dot1qConstraintVlan, dot1qConstraintSet} [Implementation] This table is always empty, in accordance with the device specifications.	Y
84	dot1qConstraintVlan {dot1qLearningConstraintsEntry 1}	NA	[Standard] A VLAN constrained by the entry. [Implementation] This table is always empty, in accordance with the device specifications.	Y
85	dot1qConstraintSet {dot1qLearningConstraintsEntry 2}	NA	[Standard] The constraint set identifier (0-65535). [Implementation] This table is always empty, in accordance with the device specifications.	Y
86	dot1qConstraintType {dot1qLearningConstraintsEntry 3}	R/NC	[Standard] The constraint type: ● independent (1) ● shared (2) [Implementation] This table is always empty, in accordance with the device specifications.	Y
87	dot1qConstraintStatus {dot1qLearningConstraintsEntry 4}	R/NC	[Standard] The constraint status. [Implementation] This table is always empty, in accordance with the device specifications.	Y
88	dot1qConstraintSetDefault {dot1qVlan 9}	R/NW	[Standard] The default value (0-65535) of the constraint set. [Implementation] 0	Y
89	dot1qConstraintTypeDefault {dot1qVlan 10}	R/NW	[Standard] The constraint set type: ● independent (1) ● shared (2) [Implementation] Fixed value of independent (1).	Y

#:

If this information is collected immediately after executing the `clear mac-address-table` operation command, the information in mac-address-table might not be shown as cleared.

2.12 ifMIB group (Interfaces Group MIB)

The ifMIB group is described in the following document:

- RFC 2233 (November 1997)

2.12.1 ifMIB

(1) ID

```
ifMIB OBJECT IDENTIFIER ::= {mib-2 31}

ifMIBObjects OBJECT IDENTIFIER ::= {ifMIB 1}
Object ID value: 1.3.6.1.2.1.31.1
```

(2) Implementation specifications

The following table lists the implementation specifications of the ifMIB group when using Ethernet.

Table 2-20 ifMIB group implementation specifications (when Ethernet is used)

#	Object identifier	Access	Implementation specifications	Implemented
1	ifXTable {ifMIBObjects 1}	NA	[Standard] A table of objects that are added to an interface entity. [Implementation] Same as the standard.	Y
2	ifXEntry {ifXTable 1}	NA	[Standard] A list of interface information that is added. AUGMENTS {ifEntry} [Implementation] Same as the standard.	Y
3	ifName {ifXEntry 1}	R/O	[Standard] The name of the interface. [Implementation] Fixed value for each interface.	Y
4	ifInMulticastPkts {ifXEntry 2}	R/O	[Standard] The number of multicast packets sent to the upper layer protocol. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
5	ifInBroadcastPkts {ifXEntry 3}	R/O	[Standard] The number of broadcast packets sent to the upper layer protocol. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
6	ifOutMulticastPkts {ifXEntry 4}	R/O	[Standard] The number of multicast packets sent by the upper layer. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
7	ifOutBroadcastPkts {ifXEntry 5}	R/O	[Standard] The number of broadcast packets sent by the upper layer. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
8	ifHCInOctets {ifXEntry 6}	R/O	[Standard] The number of octets received by this interface. A 64-bit version of ifInOctets. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Total number of received octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Total number of received octets between the DA and FCS fields in the MAC header. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
9	ifHCInUcastPkts {ifXEntry 7}	R/O	[Standard] The number of unicast packets sent to the upper protocol. A 64-bit version of ifInUcastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
10	ifHCInMulticastPkts {ifXEntry 8}	R/O	[Standard] The number of multicast packets sent to the upper layer protocol. A 64-bit version of ifInMulticastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
11	ifHCInBroadcastPkts {ifXEntry 9}	R/O	[Standard] The number of broadcast packets sent to the upper layer protocol. A 64-bit version of ifInBroadcastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
12	ifHCOutOctets {ifXEntry 10}	R/O	[Standard] The number of octets sent by this interface. A 64-bit version of ifOutOctets. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Total number of transmitted octets between the DA and FCS fields in MAC headers. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Total number of transmitted octets between the DA and FCS fields in MAC headers. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
13	ifHCOutUcastPkts {ifXEntry 11}	R/O	[Standard] The number of unicast packets sent by the upper layer. A 64-bit version of ifOutUcastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
14	ifHCOutMulticast Pkts {ifXEntry 12}	R/O	[Standard] The number of multicast packets sent by the upper layer. A 64-bit version of ifOutMulticastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
15	ifHCOutBroadcast Pkts {ifXEntry 13}	R/O	[Standard] The number of broadcast packets sent by the upper layer. A 64-bit version of ifOutBroadcastPkts. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of 0.	Y
16	ifLinkUpDownTra pEnable {ifXEntry 14}	R/NW	[Standard] Indicates whether this interface sends LinkUp and LinkDown traps: ● enable (1) ● disable (2) [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Same as the standard. ● When the ifIndex indicates a VLAN: Same as the standard. ● When the ifIndex indicates a link aggregation: Same as the standard. ● When the ifIndex indicates a peer-link: Fixed value of disable (2).	Y

#	Object identifier	Access	Implementation specifications	Implemented
17	ifHighSpeed {ifXEntry 15}	R/O	[Standard] The current line speed (in Mbit/s) of this interface. Fractions are rounded to the nearest megabit per second. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: If no bandwidth is set by using a configuration command, the line speed of the interface in question is shown. If a bandwidth has been set, the set bandwidth is shown. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Shows the sum of the line speed values of the physical ports used by the link aggregation. ● When the ifIndex indicates a peer-link: The sum of the line speed values of the physical ports used by the peer-link.	Y
18	ifPromiscuousMode {ifXEntry 16}	R/O	[Standard] Receive mode: ● true (1) ● false (2) [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: true (1) ● When the ifIndex indicates a VLAN: false (2) ● When the ifIndex indicates a link aggregation: true (1) ● When the ifIndex indicates a peer-link: true (1).	Y
19	ifConnectorPresent {ifXEntry 17}	R/O	[Standard] The connection status of the physical line: {true (1), false (2)} [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: true (1) ● When the ifIndex indicates a VLAN: false (2) ● When the ifIndex indicates a link aggregation: false (2) ● When the ifIndex indicates a peer-link: false (2).	Y
20	ifAlias {ifXEntry 18}	R/NW	[Standard] An alias name defined by Network Manager. [Implementation] Additional information that is set for each interface during configuration.	Y

#	Object identifier	Access	Implementation specifications	Implemented
21	ifCounterDiscontinuityTime {ifXEntry 19}	R/O	[Standard] The sysUpTime when the counter was discontinued. [Implementation] Depends on the interface as follows: ● When the ifIndex indicates a port: Fixed value of 0. ● When the ifIndex indicates a VLAN: Fixed value of 0. ● When the ifIndex indicates a link aggregation: Fixed value of 0. ● When the ifIndex indicates a peer-link: Fixed value of 0.	M

2.13 IEEE 8021-CFM-MIB group

2.13.1 dot1agCfmStack group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmStack OBJECT IDENTIFIER ::= {dot1agMIBObjects 1}

dot1agCfmStackTable OBJECT IDENTIFIER ::= {dot1agCfmStack 1}
Object ID value: 1.3.111.2.802.1.1.8.1.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmStack group.

Table 2-21 dot1agCfmStack implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmStackTable {dot1agCfmStack 1 }	NA	[Standard] Interface information that is specified for an MP. [Implementation] Same as the standard.	Y
2	dot1agCfmStackEntry {dot1agCfmStack Table 1 }	NA	[Standard] A stack table entry. INDEX { dot1agCfmStackifIndex, dot1agCfmStackVlanIdOrNone, dot1agCfmStackMdLevel, dot1agCfmStackDirection } [Implementation] Same as the standard.	Y
3	dot1agCfmStackifIndex {dot1agCfmStack Entry 1}	NA	[Standard] Indicates a port at the MEP. [Implementation] Same as the standard.	Y
4	dot1agCfmStackVlanIdOrNone { dot1agCfmStack Entry 2}	NA	[Standard] A VLAN ID assigned to the MP. [Implementation] Same as the standard.	Y
5	dot1agCfmStack MdLevel { dot1agCfmStack Entry 3}	NA	[Standard] The domain level of the MP. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	dot1agCfmStackDirection { dot1agCfmStackEntry 4}	NA	[Standard] The direction of the MP. [Implementation] Same as the standard.	Y
7	dot1agCfmStackMdIndex { dot1agCfmStackEntry 5}	R/O	[Standard] The domain index in dot1agCfmMdTable. [Implementation] Same as the standard.	Y
8	dot1agCfmStackMaIndex {dot1agCfmStackEntry 6}	R/O	[Standard] The MA index in dot1agCfmMaNetTable and dot1agCfmMaCompTable. [Implementation] Same as the standard.	Y
9	dot1agCfmStackMepId {dot1agCfmStackEntry 7}	R/O	[Standard] The MEP ID [Implementation] Same as the standard.	Y
10	dot1agCfmStackMacAddress { dot1agCfmStackEntry 8}	R/O	[Standard] The MAC address of the MP. [Implementation] Same as the standard.	Y

2.13.2 dot1agCfmVlan group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmVlan OBJECT IDENTIFIER ::= {dot1agMIBObjects 3}

dot1agCfmVlanTable OBJECT IDENTIFIER ::= {dot1agCfmVlan 1}
Object ID value: 1.3.111.2.802.1.1.8.1.3.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmVlan group.

Table 2-22 dot1agCfmVlan implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmVlanTable { dot1agCfmVlan 1 }	NA	[Standard] Defines a VLAN association. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
2	dot1agCfmVlanEntry { dot1agCfmVlanTable 1 }	NA	[Standard] A VLAN table entry. INDEX { dot1agCfmVlanComponentId, dot1agCfmVlanVid } [Implementation] Same as the standard.	Y
3	dot1agCfmVlanComponentId { dot1agCfmVlanEntry 1}	NA	[Standard] A component in the system to which the dot1agCfmVlanEntry information is to be applied. [Implementation] Same as the standard.	Y
4	dot1agCfmVlanVid { dot1agCfmVlanEntry 2 }	NA	[Standard] A VLAN in the MA VLAN group. This is not the primary VLAN. [Implementation] Same as the standard.	Y
5	dot1agCfmVlanPrimaryVid { dot1agCfmVlanEntry 3 }	R/NC	[Standard] The primary VLAN ID. [Implementation] Same as the standard. Read_Only.	Y
6	dot1agCfmVlanRowStatus { dot1agCfmVlanEntry 4 }	R/NC	[Standard] The table status: ● active (1) ● notInService (2) [Implementation] Same as the standard. Read_Only.	Y

2.13.3 dot1agCfmMd group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}

dot1agCfmMd OBJECT IDENTIFIER ::= {dot1agMIBObjects 5}
Object ID value: 1.3.111.2.802.1.1.8.1.5

dot1agCfmMdTable OBJECT IDENTIFIER ::= {dot1agCfmMd 2}
Object ID value: 1.3.111.2.802.1.1.8.1.5.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMd group.

Table 2-23 dot1agCfmMd implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMdTableNextIndex { dot1agCfmMd 1 }	R/O	[Standard] An index used to generate a dot1agCfmMdTable. [Implementation] Fixed value of 0.	Y
2	dot1agCfmMdTable { dot1agCfmMd 2 }	NA	[Standard] A domain table. [Implementation] Same as the standard.	Y
3	dot1agCfmMdEntry { dot1agCfmMdTable 1 }	NA	[Standard] An entry for a domain table. INDEX {dot1agCfmMdIndex } [Implementation] Same as the standard.	Y
4	dot1agCfmMdIndex { dot1agCfmMdEntry 1 }	NA	[Standard] A domain table index. [Implementation] Same as the standard.	Y
5	dot1agCfmMdFormat { dot1agCfmMdEntry 2 }	R/NC	[Standard] Domain name type: ● none (1) ● dnsLikeName (2) ● macAddressAndUint (3) ● charString (4) [Implementation] Same as the standard. Read_Only.	Y
6	dot1agCfmMdName { dot1agCfmMdEntry 3 }	R/NC	[Standard] Domain name. [Implementation] Same as the standard. Read_Only.	Y
7	dot1agCfmMdMdLevel { dot1agCfmMdEntry 4 }	R/NC	[Standard] Domain level. [Implementation] Same as the standard. Read_Only.	Y
8	dot1agCfmMdMhfCreation { dot1agCfmMdEntry 5 }	R/NC	[Standard] Indicates whether a MHF (MIP) can be generated: ● defMHFnone (1) ● defMHFdefault (2) ● defMHFexplicit (3) [Implementation] Fixed value of defMHFexplicit (3). Read_Only.	Y
9	dot1agCfmMdMhfIdPermission { dot1agCfmMdEntry 6 }	R/NC	[Standard] A value contained in the Sender ID TLV: ● sendIdNone (1) ● sendIdChassis (2) ● sendIdManage (3) ● sendIdChassisManage (4) [Implementation] Fixed value of sendIdChassis (2). Read_Only.	Y

#	Object identifier	Access	Implementation specifications	Implemented
10	dot1agCfmMdMaNextIndex { dot1agCfmMdEntry 7 }	R/O	[Standard] An index value used to generate dot1agCfmMaNetTable and dot1agCfmMaCompTable. [Implementation] Fixed value of 0.	Y
11	dot1agCfmMdRowStatus { dot1agCfmMdEntry 8 }	R/NC	[Standard] The table status: ● active (1) ● notInService (2) [Implementation] Fixed value of active (1). Read_Only.	Y

2.13.4 dot1agCfmMaNet group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMa OBJECT IDENTIFIER ::= {dot1agMIBObjects 6}

dot1agCfmMaNetTable OBJECT IDENTIFIER ::= {dot1agCfmMa 1}
Object ID value: 1.3.111.2.802.1.1.8.1.6.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMaNet group.

Table 2-24 dot1agCfmMaNet implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMaNetTable { dot1agCfmMa 1 }	NA	[Standard] An MA table. [Implementation] Same as the standard.	Y
2	dot1agCfmMaNetEntry { dot1agCfmMaNetTable 1 }	NA	[Standard] An MA table entry. INDEX {dot1agCfmMdIndex, dot1agCfmMaIndex } [Implementation] Same as the standard.	Y
3	dot1agCfmMaIndex { dot1agCfmMaNetEntry 1}	NA	[Standard] The MA table index. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
4	dot1agCfmMaNetFormat { dot1agCfmMaNetEntry 2 }	R/NC	[Standard] The MA name type: ● ieeeReserved (0) ● primaryVid (1) ● charString (2) ● unsignedInt16 (3) ● rfc2865VpnId (4) [Implementation] This Switch returns a value from 1 to 3. Read_Only.	Y
5	dot1agCfmMaNetName { dot1agCfmMaNetEntry 3 }	R/NC	[Standard] An MA name. [Implementation] Same as the standard. Read_Only.	Y
6	dot1agCfmMaNetCcmInterval { dot1agCfmMaNetEntry 4 }	R/NC	[Standard] The time interval between CCM transmissions: ● intervalInvalid (0) ● interval300Hz (1) ● interval10ms (2) ● interval100ms (3) ● interval1s (4) ● interval10s (5) ● interval1min (6) ● interval10min (7) [Implementation] This Switch returns a value from 4 to 7. Read_Only.	Y
7	dot1agCfmMaNetRowStatus { dot1agCfmMaNetEntry 5 }	R/NC	[Standard] The table status: ● active (1) ● notInService (2) [Implementation] Fixed value of active (1). Read_Only.	Y

2.13.5 dot1agCfmMaComp group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMa OBJECT IDENTIFIER ::= {dot1agMIBObjects 6}

dot1agCfmMaCompTable OBJECT IDENTIFIER ::= {dot1agCfmMa 2}
Object ID value: 1.3.111.2.802.1.1.8.1.6.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMaComp group.

Table 2-25 dot1agCfmMaComp implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMaCompTable { dot1agCfmMa 2 }	NA	[Standard] An MA table. [Implementation] Same as the standard.	Y
2	dot1agCfmMaCompEntry { dot1agCfmMaCompTable 1 }	NA	[Standard] An MA table entry. INDEX {dot1agCfmMaComponentId, dot1agCfmMdIndex, dot1agCfmMaIndex } [Implementation] Same as the standard.	Y
3	dot1agCfmMaComponentId { dot1agCfmMaCompEntry 1 }	NA	[Standard] A component in the system to which the `dot1agCfmMaCompEntry` information is to be applied. [Implementation] Same as the standard.	Y
4	dot1agCfmMaCompPrimaryVlanId { dot1agCfmMaCompEntry 2 }	R/NC	[Standard] The primary VLAN ID. [Implementation] Same as the standard. `Read_Only`.	Y
5	dot1agCfmMaCompMhfCreation { dot1agCfmMaCompEntry 3 }	R/NC	[Standard] The MIP generation condition in the MA: ● defMHFnone (1) ● defMHFdefault (2) ● defMHFexplicit (3) ● defMHFdefer (4) [Implementation] defMHFexplicit (3) for this Switch. `Read_Only`.	Y
6	dot1agCfmMaCompIdPermission { dot1agCfmMaCompEntry 4 }	R/NC	[Standard] The Sender ID TLV: ● sendIdNone (1) ● sendIdChassis (2) ● sendIdManage (3) ● sendIdChassisManage (4) [Implementation] Fixed value of sendIdChassis (2). `Read_Only`.	Y
7	dot1agCfmMaCompNumberOfVids { dot1agCfmMaCompEntry 5 }	R/NC	[Standard] The number of VLANs in the MA. [Implementation] Same as the standard. `Read_Only`.	Y
8	dot1agCfmMaCompRowStatus { dot1agCfmMaCompEntry 6 }	R/NC	[Standard] The table status: ● active (1) ● notInService (2) [Implementation] Fixed value of active (1). `Read_Only`.	Y

2.13.6 dot1agCfmMaMepList group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
```

```
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMa OBJECT IDENTIFIER ::= {dot1agMIBObjects 6}
dot1agCfmMaMepListTable OBJECT IDENTIFIER ::= {dot1agCfmMa 3}
Object ID value: 1.3.111.2.802.1.1.8.1.6.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMaMepList group.

Table 2-26 dot1agCfmMaMepList implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMaMepListTable { dot1agCfmMa 3 }	NA	[Standard] A list of MEP IDs that belong to an MA. [Implementation] Same as the standard.	Y
2	dot1agCfmMaMepListEntry { dot1agCfmMaMepListTable 1 }	NA	[Standard] A MEP table entry. INDEX { dot1agCfmMdIndex, dot1agCfmMaIndex, dot1agCfmMaMepListIdentifier} [Implementation] Same as the standard.	Y
3	dot1agCfmMaMepListIdentifier { dot1agCfmMaMepListEntry 1 }	NA	[Standard] The MEP ID [Implementation] Same as the standard.	Y
4	dot1agCfmMaMepListRowStatus { dot1agCfmMaMepListEntry 2 }	R/NC	[Standard] The table status: ● active (1) ● notInService (2) [Implementation] Same as the standard. Read_Only.	Y

2.13.7 dot1agCfmMep group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMep OBJECT IDENTIFIER ::= {dot1agMIBObjects 7}

dot1agCfmMepTable OBJECT IDENTIFIER ::= {dot1agCfmMep 1}
Object ID value: 1.3.111.2.802.1.1.8.1.7.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMep group.

Table 2-27 dot1agCfmMep implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMepTable { dot1agCfmMep 1 }	NA	[Standard] A MEP table. [Implementation] Same as the standard.	Y
2	dot1agCfmMepEntry { dot1agCfmMepTable 1 }	NA	[Standard] A MEP table entry. INDEX { dot1agCfmMdIndex, dot1agCfmMaIndex, dot1agCfmMepIdentifier } [Implementation] Same as the standard.	Y
3	dot1agCfmMepIdentifier { dot1agCfmMepEntry 1}	NA	[Standard] The MEP ID. [Implementation] Same as the standard.	Y
4	dot1agCfmMepIfIndex { dot1agCfmMepEntry 2 }	R/NC	[Standard] The ifIndex of an interface for which the MEP is defined. [Implementation] Same as the standard. Read_Only.	Y
5	dot1agCfmMepDirection { dot1agCfmMepEntry 3 }	R/NC	[Standard] The MEP direction: ● down (1) ● up (2) [Implementation] Same as the standard. Read_Only.	Y
6	dot1agCfmMepPrimaryVid { dot1agCfmMepEntry 4 }	R/NC	[Standard] The primary VLAN ID of the MEP. [Implementation] Same as the standard. Read_Only.	Y
7	dot1agCfmMepActive { dot1agCfmMepEntry 5 }	R/NC	[Standard] The MEP status. ● true (1) ● false (2) [Implementation] Same as the standard. Read_Only.	Y
8	dot1agCfmMepFngState { dot1agCfmMepEntry 6 }	R/O	[Standard] The MEP fault status: ● fngReset (1) ● fngDefect (2) ● fngReportDefect (3) ● fngDefectReported (4) ● fngDefectClearing (5) [Implementation] Same as the standard.	Y
9	dot1agCfmMepCciEnabled { dot1agCfmMepEntry 7 }	R/NC	[Standard] A value of true indicates that a CCM is to be generated. ● true (1) ● false (2) [Implementation] Same as the standard. Read_Only.	Y

#	Object identifier	Access	Implementation specifications	Implemented
10	dot1agCfmMepCcmLtmPriority { dot1agCfmMepEntry 8 }	R/NC	[Standard] The priority for CCMs and linktrace messages. [Implementation] Same as the standard. Read_Only.	Y
11	dot1agCfmMepMacAddress { dot1agCfmMepEntry 9 }	R/O	[Standard] The MAC address of the MEP. [Implementation] Same as the standard.	Y
12	dot1agCfmMepLowPrDef { dot1agCfmMepEntry 10}	R/NC	[Standard] The lowest error priority: ● allDef (1) ● macRemErrXcon (2) ● remErrXcon (3) ● errXcon (4) ● xcon (5) ● noXcon (6) [Implementation] Same as the standard. Read_Only.	Y
13	dot1agCfmMepFngAlarmTime { dot1agCfmMepEntry 11 }	R/NC	[Standard] The time an error must be present before an error alarm is issued. [Implementation] Same as the standard. Read_Only.	Y
14	dot1agCfmMepFngResetTime { dot1agCfmMepEntry 12 }	R/NC	[Standard] The time an error must be absent before the error alarm is reset. [Implementation] Same as the standard. Read_Only.	Y
15	dot1agCfmMepHighestPrDefect { dot1agCfmMepEntry 13 }	R/O	[Standard] The highest error priority in the MEP: ● none (0) ● defRDICCM (1) ● defMACstatus (2) ● defRemoteCCM (3) ● defErrorCCM (4) ● defXconCCM (5) [Implementation] Same as the standard.	Y
16	dot1agCfmMepDefects { dot1agCfmMepEntry 14 }	R/O	[Standard] The bit value that indicates each error: ● bDefRDICCM (0) ● bDefMACstatus (1) ● bDefRemoteCCM (2) ● bDefErrorCCM (3) ● bDefXconCCM (4) [Implementation] Same as the standard.	Y
17	dot1agCfmMepErrorCcmLastFailure { dot1agCfmMepEntry 15 }	R/O	[Standard] The last CCM received due to a DefErrorCCM error. [Implementation] Same as the standard, but no more than 58 bytes of a CFM PDU.	Y

#	Object identifier	Access	Implementation specifications	Implemented
18	dot1agCfmMepXconCcmLastFailure { dot1agCfmMepEntry 16 }	R/O	[Standard] The last CCM received due to a DefXconCCM error. [Implementation] Same as the standard, but no more than 58 bytes of a CFM PDU.	Y
19	dot1agCfmMepCcmSequenceErrors { dot1agCfmMepEntry 17 }	R/O	[Standard] The total number of out-of-sequence CCMs. [Implementation] Same as the standard.	Y
20	dot1agCfmMepCciSentCcms { dot1agCfmMepEntry 18 }	R/O	[Standard] The total number of CC messages transmitted. [Implementation] Same as the standard.	Y
21	dot1agCfmMepNextLbmTransId { dot1agCfmMepEntry 19 }	R/O	[Standard] The next sequence number in a loopback message. [Implementation] Same as the standard.	Y
22	dot1agCfmMepLbrIn { dot1agCfmMepEntry 20 }	R/O	[Standard] The number of loopback replies received. [Implementation] Same as the standard.	Y
23	dot1agCfmMepLbrInOutOfOrder { dot1agCfmMepEntry 21 }	R/O	[Standard] The number of out-of-order loopback replies. [Implementation] Same as the standard.	Y
24	dot1agCfmMepLbrBadMsdu { dot1agCfmMepEntry 22}	R/O	[Standard] The total number of loopback replies received that contain a mismatched mac_service_data_unit value. [Implementation] Same as the standard.	Y
25	dot1agCfmMepLtmNextSeqNumber { dot1agCfmMepEntry 23 }	R/O	[Standard] The next forwarding ID in a link layer message. [Implementation] Same as the standard.	Y
26	dot1agCfmMepUnexpLtrIn { dot1agCfmMepEntry 24 }	R/O	[Standard] The number of unexpected linktrace replies received. [Implementation] Same as the standard.	Y
27	dot1agCfmMepLbrOut { dot1agCfmMepEntry 25 }	R/O	[Standard] The number of transmitted loopback replies received. [Implementation] Same as the standard.	Y
28	dot1agCfmMepTransmitLbmStatus { dot1agCfmMepEntry 26 }	R/NC	[Standard] Indicates whether a loopback message is to be transmitted. ● true (1) ● false (2) [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
29	dot1agCfmMepTransmitLbmDestMacAddress { dot1agCfmMepEntry 27 }	R/NC	[Standard] The destination MAC address of the loopback message. This is valid when #31 is set to `false`. [Implementation] Same as the standard. `Read_Only`.	Y
30	dot1agCfmMepTransmitLbmDestMepId { dot1agCfmMepEntry 28 }	R/NC	[Standard] The destination MEP ID of a loopback message. This is valid when #31 is set to `true`. [Implementation] Not supported because #31 is set to `false` in this system.	N
31	dot1agCfmMepTransmitLbmDestIsMepId {dot1agCfmMepEntry 29 }	R/NC	[Standard] true (1): The MEP ID is used for loopback transmission. false (2): The MEP destination MAC address is used for loopback transmission. [Implementation] Fixed value of `false`. `Read_Only`.	Y
32	dot1agCfmMepTransmitLbmMessages {dot1agCfmMepEntry 30 }	R/NC	[Standard] The number of loopback messages to be sent. [Implementation] Same as the standard. `Read_Only`.	Y
33	dot1agCfmMepTransmitLbmDataTlv { dot1agCfmMepEntry 31 }	R/NC	[Standard] Data TLV data. [Implementation] Same as the standard. `Read_Only`.	Y
34	dot1agCfmMepTransmitLbmVlanPriority { dot1agCfmMepEntry 32 }	R/NC	[Standard] The priority to be used in a VLAN tag. [Implementation] Same as the standard. `Read_Only`.	Y
35	dot1agCfmMepTransmitLbmVlanDropEnable { dot1agCfmMepEntry 33 }	R/NC	[Standard] A Drop Enable bit value in a VLAN tag. ● true (1) ● false (2) [Implementation] Fixed value of `false`. `Read_Only`.	Y
36	dot1agCfmMepTransmitLbmResultOK { dot1agCfmMepEntry 34 }	R/O	[Standard] The operation result. ● true (1) ● false (2) [Implementation] Fixed value of true (1).	Y
37	dot1agCfmMepTransmitLbmSeqNumber { dot1agCfmMepEntry 35 }	R/O	[Standard] The loopback transaction ID (dot1agCfmMepNextLbmTransId) of the first loopback message sent. [Implementation] The loopback transaction ID of the previous loopback message sent.	Y

#	Object identifier	Access	Implementation specifications	Implemented
38	dot1agCfmMepTransmitLtmStatus { dot1agCfmMepEntry 36 }	R/NC	[Standard] The transmission status of the linktrace message. [Implementation] Same as the standard.	Y
39	dot1agCfmMepTransmitLtmFlags { dot1agCfmMepEntry 37 }	R/NC	[Standard] The flag for the linktrace message sent by the MEP. [Implementation] Fixed value of 0. Read_Only.	Y
40	dot1agCfmMepTransmitLtmTargetMacAddress { dot1agCfmMepEntry 38 }	R/NC	[Standard] The destination MAC address of the linktrace message. This is valid when #42 is set to false. [Implementation] Same as the standard. Read_Only.	Y
41	dot1agCfmMepTransmitLtmTargetMepId { dot1agCfmMepEntry 39 }	R/NC	[Standard] The destination MEP ID of the linktrace message. This is valid when #42 is set to true. [Implementation] Not supported because #42 is set to false in this system.	N
42	dot1agCfmMepTransmitLtmTargetIsMepId { dot1agCfmMepEntry 40 }	R/NC	[Standard] ● true (1): Destination MEP ID ● false (2): Destination MAC address [Implementation] Fixed value of false. Read_Only.	Y
43	dot1agCfmMepTransmitLtmTtl { dot1agCfmMepEntry 41 }	R/NC	[Standard] The TTL in the linktrace message. [Implementation] Same as the standard. Read_Only.	Y
44	dot1agCfmMepTransmitLtmResult { dot1agCfmMepEntry 42 }	R/O	[Standard] The operation result. ● true (1) ● false (2) [Implementation] Fixed value of true (1).	Y
45	dot1agCfmMepTransmitLtmSeqNumber { dot1agCfmMepEntry 43 }	R/O	[Standard] The ID of a linktrace message that was sent. [Implementation] Same as the standard.	Y
46	dot1agCfmMepTransmitLtmEgressIdentifier { dot1agCfmMepEntry 44 }	R/NC	[Standard] The linktrace message transaction identifier of the linktrace message to be sent. [Implementation] Same as the standard. Read_Only.	Y
47	dot1agCfmMepRowStatus { dot1agCfmMepEntry 45 }	R/NC	[Standard] The table status. ● active (1) ● notInService (2) [Implementation] Same as the standard. Read_Only.	Y

2.13.8 dot1agCfmLtr group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMep OBJECT IDENTIFIER ::= {dot1agMIBObjects 7}

dot1agCfmLtrTable OBJECT IDENTIFIER ::= {dot1agCfmMep 2}
Object ID value: 1.3.111.2.802.1.1.8.1.7.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmLtr group.

Table 2-28 dot1agCfmLtr implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmLtrTable { dot1agCfmMep 2 }	NA	[Standard] Linktrace reply lists. [Implementation] Same as the standard.	Y
2	dot1agCfmLtrEntry { dot1agCfmLtrTable 1 }	NA	[Standard] A table entry for a linktrace reply list. INDEX { dot1agCfmMdIndex, dot1agCfmMaIndex, dot1agCfmMepIdentifier, dot1agCfmLtrSeqNumber, dot1agCfmLtrReceiveOrder } [Implementation] Same as the standard.	Y
3	dot1agCfmLtrSeqNumber { dot1agCfmLtrEntry 1}	NA	[Standard] The ID of a linktrace reply list. [Implementation] Same as the standard.	Y
4	dot1agCfmLtrReceiveOrder { dot1agCfmLtrEntry 2 }	NA	[Standard] An identifier used to distinguish between two or more linktrace replies. [Implementation] Same as the standard.	Y
5	dot1agCfmLtrTtl { dot1agCfmLtrEntry 3 }	R/O	[Standard] The TTL for the linktrace reply. [Implementation] Same as the standard.	Y
6	dot1agCfmLtrForwarded { dot1agCfmLtrEntry 4 }	R/O	[Standard] Indicates whether the reply was transmitted by an MP. ● true (1) ● false (2) [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
7	dot1agCfmLtrTerminalMep { dot1agCfmLtrEntry 5 }	R/O	[Standard] Indicates whether the transmitted linktrace message reached the MEP in the MA. ● true (1) ● false (2) [Implementation] Same as the standard.	Y
8	dot1agCfmLtrLastEgressIdentifier { dot1agCfmLtrEntry 6 }	R/O	[Standard] The last Egress ID. [Implementation] Same as the standard.	Y
9	dot1agCfmLtrNextEgressIdentifier { dot1agCfmLtrEntry 7 }	R/O	[Standard] The next Egress ID. [Implementation] Same as the standard.	Y
10	dot1agCfmLtrRelay { dot1agCfmLtrEntry 8 }	R/O	[Standard] The value of the relay action field: ● rlyHit (1) ● rlyFdb (2) ● rlyMpdb (3) [Implementation] Same as the standard.	Y
11	dot1agCfmLtrChassisIdSubtype { dot1agCfmLtrEntry 9 }	R/O	[Standard] The value of the chassis format: ● chassisComponent (1) ● interfaceAlias (2) ● portComponent (3) ● macAddress (4) ● networkAddress (5) ● interfaceName (6) ● local (7) [Implementation] Same as the standard.	Y
12	dot1agCfmLtrChassisId { dot1agCfmLtrEntry 10 }	R/O	[Standard] The chassis ID of the Sender ID TLV. [Implementation] Same as the standard.	Y
13	dot1agCfmLtrManAddressDomain { dot1agCfmLtrEntry 11 }	R/O	[Standard] The TDomain. [Implementation] Same as the standard.	Y
14	dot1agCfmLtrManAddress { dot1agCfmLtrEntry 12 }	R/O	[Standard] The address of the SNMP Agent. [Implementation] Same as the standard. No more than 16 bytes.	Y
15	dot1agCfmLtrIngress { dot1agCfmLtrEntry 13 }	R/O	[Standard] The return value in the Ingress Action field of the linktrace message. ● ingNoTlv (0) ● ingOk (1) ● ingDown (2) ● ingBlocked (3) ● ingVid (4) [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
16	dot1agCfmLtrIngressMac { dot1agCfmLtrEntry 14 }	R/O	[Standard] The Ingress MAC address. [Implementation] Same as the standard.	Y
17	dot1agCfmLtrIngressPortIdSubtype { dot1agCfmLtrEntry 15 }	R/O	[Standard] The format of the physical port: ● interfaceAlias (1) ● portComponent (2) ● macAddress (3) ● networkAddress (4) ● interfaceName (5) ● agentCircuitId (6) ● local (7) [Implementation] Same as the standard.	Y
18	dot1agCfmLtrIngressPortId { dot1agCfmLtrEntry 16 }	R/O	[Standard] The Port ID. [Implementation] Same as the standard.	Y
19	dot1agCfmLtrEgress { dot1agCfmLtrEntry 17 }	R/O	[Standard] The Egress action field of the linktrace messages. ● egrNoTlv (0) ● egrOK (1) ● egrDown (2) ● egrBlocked (3) ● egrVid (4) [Implementation] Same as the standard.	Y
20	dot1agCfmLtrEgressMac { dot1agCfmLtrEntry 18 }	R/O	[Standard] The Egress MAC address field. [Implementation] Same as the standard.	Y
21	dot1agCfmLtrEgressPortIdSubtype { dot1agCfmLtrEntry 19 }	R/O	[Standard] The format of the Egress Port ID: ● interfaceAlias (1) ● portComponent (2) ● macAddress (3) ● networkAddress (4) ● interfaceName (5) ● agentCircuitId (6) ● local (7) [Implementation] Same as the standard.	Y
22	dot1agCfmLtrEgressPortId { dot1agCfmLtrEntry 20 }	R/O	[Standard] The Egress Port ID. [Implementation] Same as the standard.	Y
23	dot1agCfmLtrOrganizationSpecificTlv { dot1agCfmLtrEntry 21 }	R/O	[Standard] OUI of Organization-Specific TLV. [Implementation] Same as the standard, but no more than 30 bytes.	Y

2.13.9 dot1agCfmMepDb group

(1) ID

```
org OBJECT IDENTIFIER ::= {iso 3}
ieee OBJECT IDENTIFIER ::= {org 111}
standards-association-numbered-series-standards OBJECT IDENTIFIER ::= {ieee
2}
lan-man-stds OBJECT IDENTIFIER ::=
{standards-association-numbered-series-standards 802}
ieee802dot1 OBJECT IDENTIFIER ::= {lan-man-stds 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021CfmMib OBJECT IDENTIFIER ::= {ieee802dot1mibs 8}
dot1agMIBObjects OBJECT IDENTIFIER ::= {ieee8021CfmMib 1}
dot1agCfmMep OBJECT IDENTIFIER ::= {dot1agMIBObjects 7}

dot1agCfmMepDbTable OBJECT IDENTIFIER ::= {dot1agCfmMep 3}
Object ID value: 1.3.111.2.802.1.1.8.1.7.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot1agCfmMepDb group.

Table 2-29 dot1agCfmMepDb implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1agCfmMepDbTable { dot1agCfmMep 3 }	NA	[Standard] A MEP database table. [Implementation] Same as the standard.	Y
2	dot1agCfmMepDbEntry { dot1agCfmMepDbTable 1 }	NA	[Standard] A MEP database table entry. INDEX { dot1agCfmMdIndex, dot1agCfmMaIndex, dot1agCfmMepIdentifier, dot1agCfmMepDbRMepIdentifier } [Implementation] Same as the standard.	Y
3	dot1agCfmMepDbRMepIdentifier { dot1agCfmMepDbEntry 1 }	NA	[Standard] The MEP ID of the remote MEP. [Implementation] Same as the standard.	Y
4	dot1agCfmMepDbRMepState { dot1agCfmMepDbEntry 2}	R/O	[Standard] The operation state of the remote MEP: ● rMepIdle (1) ● rMepStart (2) ● rMepFailed (3) ● rMepOk (4) [Implementation] Same as the standard.	Y
5	dot1agCfmMepDbRMepFailedOkTime { dot1agCfmMepDbEntry 3 }	R/O	[Standard] The time elapsed since the remote MEP last changed to Fail or OK. [Implementation] Same as the standard.	Y
6	dot1agCfmMepDbMacAddress { dot1agCfmMepDbEntry 4 }	R/O	[Standard] The MAC address of the remote MEP. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
7	dot1agCfmMepDbRdi { dot1agCfmMepDbEntry 5 }	R/O	[Standard] The RDI bit of the last CCM that was received. [Implementation] Same as the standard.	Y
8	dot1agCfmMepDbPortStatusTlv { dot1agCfmMepDbEntry 6}	R/O	[Standard] The TLV port state of the last CCM that was received from a remote MEP: ● psNoPortStateTLV (0) ● psBlocked (1) ● psUp (2) [Implementation] Same as the standard.	Y
9	dot1agCfmMepDbInterfaceStatusTlv { dot1agCfmMepDbEntry 7}	R/O	[Standard] The TLV interface state of the last CCM that was received from a remote MEP: ● isNoInterfaceStatusTLV (0) ● isUp (1) ● isDown (2) ● isTesting (3) ● isUnknown (4) ● isDormant (5) ● isNotPresent (6) ● isLowerLayerDown (7) [Implementation] Same as the standard.	Y
10	dot1agCfmMepDbChassisIdSubtype { dot1agCfmMepDbEntry 8 }	R/O	[Standard] The format of the chassis ID of the last CCM that was received: ● chassisComponent (1) ● interfaceAlias (2) ● portComponent (3) ● macAddress (4) ● networkAddress (5) ● interfaceName (6) ● local (7) [Implementation] Same as the standard.	Y
11	dot1agCfmMepDbChassisId { dot1agCfmMepDbEntry 9 }	R/O	[Standard] The chassis ID of the last CCM that was received. [Implementation] Same as the standard.	Y
12	dot1agCfmMepDbManAddressDomain { dot1agCfmMepDbEntry 10 }	R/O	[Standard] The TDomain [Implementation] Same as the standard.	Y
13	dot1agCfmMepDbManAddress { dot1agCfmMepDbEntry 11 }	R/O	[Standard] The TAddress [Implementation] Same as the standard. No more than 16 bytes.	Y

2.14 IEEE 8023-LAG-MIB group

The IEEE 8023-LAG-MIB group is described in the following document:

- IEEE 8023-LAG-MIB

2.14.1 dot3adAgg group

(1) ID

```
member-body   OBJECT IDENTIFIER ::= {iso 2}
us            OBJECT IDENTIFIER ::= {member-body 840}
ieee802dot3   OBJECT IDENTIFIER ::= {us 10006}
snmpmibs      OBJECT IDENTIFIER ::= {ieee802dot3 300}
lagMIB        OBJECT IDENTIFIER ::= {snmpmibs 43}
lagMIBObjects OBJECT IDENTIFIER ::= {lagMIB 1}

dot3adAgg OBJECT IDENTIFIER ::= {lagMIBObjects 1}
Object ID value: 1.2.840.10006.300.43.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot3adAgg group.

Table 2-30 dot3adAgg group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot3adAggTable {dot3adAgg 1}	NA	[Standard] A table of Aggregators in this system. [Implementation] Same as the standard.	Y
2	dot3adAggEntry {dot3adAggTable 1}	NA	[Standard] A list of Aggregator parameters. INDEX {dot3adAggIndex} [Implementation] Same as the standard.	Y
3	dot3adAggIndex {dot3adAggEntry 1}	NA	[Standard] The number that identifies the interface. [Implementation] Same as the standard.	Y
4	dot3adAggMACAddress {dot3adAggEntry 2}	R/O	[Standard] A MAC address assigned to an Aggregator. [Implementation] Same as the standard.	Y
5	dot3adAggActorSystemPriority {dot3adAggEntry 3}	R/NW	[Standard] A priority value associated with the system ID of the Actor. [Implementation] Same as the standard.	Y
6	dot3adAggActorSystemID {dot3adAggEntry 4}	R/NW	[Standard] A unique identifier for the system. [Implementation] Same as the standard.	Y
7	dot3adAggAggregateOrIndividual {dot3adAggEntry 5}	R/O	[Standard] Indicates whether the Aggregator performs link aggregation or treats links individually. [Implementation] Same as the standard.	Y
8	dot3adAggActorAdminKey {dot3adAggEntry 6}	R/NW	[Standard] The current administrative key for the Aggregator. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
9	dot3adAggActorOperKey {dot3adAggEntry 7}	R/O	[Standard] The current operational key for the Aggregator. [Implementation] Same as the standard.	Y
10	dot3adAggPartnerSystemID {dot3adAggEntry 8}	R/O	[Standard] A MAC address that is the unique identifier of the current protocol partner for the Aggregator. [Implementation] Same as the standard.	Y
11	dot3adAggPartnerSystemPriority {dot3adAggEntry 9}	R/O	[Standard] A priority value associated with the system ID of the Partner. [Implementation] Same as the standard.	Y
12	dot3adAggPartnerOperKey {dot3adAggEntry 10}	R/O	[Standard] An operational key value for the current protocol partner of the Aggregator. [Implementation] Same as the standard.	Y
13	dot3adAggCollectorMaxDelay {dot3adAggEntry 11}	R/NW	[Standard] The maximum delay (10 microseconds) before a frame received by FrameCollector is sent from AggregatorParser to MACClient, or the maximum delay before the frame is discarded. [Implementation] Fixed value of 0.	M
14	dot3adAggPortListTable {dot3adAgg 2}	NA	[Standard] A list of AggregationPorts connected to the Aggregator. [Implementation] Same as the standard.	Y
15	dot3adAggPortListEntry {dot3adAggPortListTable 1}	NA	[Standard] A list of ports associated with the Aggregator. INDEX {dot3adAggIndex} [Implementation] Same as the standard.	Y
16	dot3adAggPortListPorts {dot3adAggPortListEntry 1}	R/O	[Standard] The full set of ports associated with the Aggregator. [Implementation] Same as the standard.	Y

2.14.2 dot3adAggPort group

(1) ID

```
member-body   OBJECT IDENTIFIER ::= {iso 2}
us            OBJECT IDENTIFIER ::= {member-body 840}
ieee802dot3   OBJECT IDENTIFIER ::= {us 10006}
snmpmibs      OBJECT IDENTIFIER ::= {ieee802dot3 300}
lagMIB        OBJECT IDENTIFIER ::= {snmpmibs 43}
lagMIBObjects OBJECT IDENTIFIER ::= {lagMIB 1}

dot3adAggPort OBJECT IDENTIFIER ::= {lagMIBObjects 2}
Object ID value: 1.2.840.10006.300.43.1.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot3adAggPort group.

Table 2-31 dot3adAggPort group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot3adAggPortTable {dot3adAggPort 1}	NA	[Standard] The Link Aggregation Control configuration information for all AggregationPorts. [Implementation] Same as the standard.	Y
2	dot3adAggPortEntry {dot3adAggPortTable 1}	NA	[Standard] A list of Link Aggregation Control configuration parameters for each AggregationPort. INDEX {dot3adAggPortIndex} [Implementation] Same as the standard.	Y
3	dot3adAggPortIndex {dot3adAggPortEntry 1 }	NA	[Standard] The number identifying the interface. [Implementation] Same as the standard.	Y
4	dot3adAggPortActorSystemPriority {dot3adAggPortEntry 2}	R/NW	[Standard] A priority value associated with the system ID of the Actor. [Implementation] Same as the standard.	Y
5	dot3adAggPortActorSystemID {dot3adAggPortEntry 3}	R/O	[Standard] A MAC address that determines the system ID value of the system for the AggregationPort. [Implementation] Same as the standard. Depends on the SML domain ID when SML is used.	Y
6	dot3adAggPortActorAdminKey {dot3adAggPortEntry 4}	R/NW	[Standard] An administrative key for the AggregationPort. [Implementation] Same as the standard.	Y
7	dot3adAggPortActorOperKey {dot3adAggPortEntry 5}	R/NW	[Standard] An operational key for the AggregationPort. [Implementation] Same as the standard.	Y
8	dot3adAggPortPartnerAdminSystemPriority {dot3adAggPortEntry 6}	R/NW	[Standard] An administrative priority value associated with the system ID of the Partner. [Implementation] Same as the standard.	Y
9	dot3adAggPortPartnerOperSystemPriority {dot3adAggPortEntry 7}	R/O	[Standard] An operational priority value associated with the system ID of the Partner. [Implementation] Same as the standard.	Y
10	dot3adAggPortPartnerAdminSystemID {dot3adAggPortEntry 8}	R/NW	[Standard] The system ID administrative value for the protocol partner of the AggregationPort. [Implementation] Same as the standard.	Y
11	dot3adAggPortPartnerOperSystemID {dot3adAggPortEntry 9}	R/O	[Standard] An administrative key value for the protocol partner. [Implementation] Same as the standard.	Y
12	dot3adAggPortPartnerAdminKey {dot3adAggPortEntry 10}	R/NW	[Standard] An administrative key value for the current protocol partner of the Aggregator. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
13	dot3adAggPortPartnerOperKey {dot3adAggPortEntry 11}	R/O	[Standard] An operational key for the protocol partner. [Implementation] Same as the standard.	Y
14	dot3adAggPortSelectedAggID {dot3adAggPortEntry 12}	R/O	[Standard] The ifIndex of the Aggregator for the AggregationPort. [Implementation] Same as the standard.	Y
15	dot3adAggPortAttachedAggID {dot3adAggPortEntry 13}	R/O	[Standard] The ifIndex of the Aggregator on which the AggregationPort is installed. [Implementation] Same as the standard.	Y
16	dot3adAggPortActorPort {dot3adAggPortEntry 14}	R/O	[Standard] A port number assigned to the AggregationPort. [Implementation] Same as the standard. Depends on the SML ID when SML is used.	Y
17	dot3adAggPortActorPortPriority {dot3adAggPortEntry 15}	R/NW	[Standard] A priority value assigned to the AggregationPort. [Implementation] Same as the standard.	Y
18	dot3adAggPortPartnerAdminPort {dot3adAggPortEntry 16}	R/NW	[Standard] A port number for managing the protocol partner. [Implementation] Same as the standard.	Y
19	dot3adAggPortPartnerOperPort {dot3adAggPortEntry 17}	R/O	[Standard] An operational port number that is assigned to the AggregationPort by its protocol partner. [Implementation] Same as the standard.	Y
20	dot3adAggPortPartnerAdminPortPriority {dot3adAggPortEntry 18}	R/NW	[Standard] An administrative port priority value for the protocol partner. [Implementation] Same as the standard.	Y
21	dot3adAggPortPartnerOperPortPriority {dot3adAggPortEntry 19}	R/O	[Standard] A priority value that is assigned to the AggregationPort by the partner. [Implementation] Same as the standard.	Y
22	dot3adAggPortActorAdminState {dot3adAggPortEntry 20}	R/NW	[Standard] An administrative `Actor_State` value sent by the Actor in LACPDUs. [Implementation] Same as the standard. In accordance with the ASN.1/BER standard, the bits are coded in the reverse order of that of the LACPDU.	Y
23	dot3adAggPortActorOperState {dot3adAggPortEntry 21}	R/O	[Standard] An operational `Actor_State` value sent by the Actor in LACPDUs. [Implementation] Same as the standard. In accordance with the ASN.1/BER standard, the bits are coded in the reverse order of that of the LACPDU.	Y

#	Object identifier	Access	Implementation specifications	Implemented
24	dot3adAggPortPartnerAdminState {dot3adAggPortEntry 22}	R/NW	[Standard] An administrative Actor_State value for the protocol partner. [Implementation] Same as the standard. In accordance with the ASN.1/BER standard, the bits are coded in the reverse order of that of the LACPDU.	Y
25	dot3adAggPortPartnerOperState {dot3adAggPortEntry 23}	R/O	[Standard] An Actor_State value sent by the protocol partner in the most recent LACPDU. [Implementation] Same as the standard. In accordance with the ASN.1/BER standard, the bits are coded in the reverse order of that of the LACPDU.	Y
26	dot3adAggPortAggregateOrIndividual {dot3adAggPortEntry 24}	R/O	[Standard] Indicates whether the AggregationPort can be aggregated or can only be operated as an individual link. [Implementation] Same as the standard.	Y
27	dot3adAggPortStatsTable {dot3adAggPort 2}	NA	[Standard] A table of Link Aggregation information for all ports. [Implementation] Same as the standard.#1	Y
28	dot3adAggPortStatsEntry {dot3adAggPortStatsTable 1}	NA	[Standard] A list of Link Aggregation control protocol statistics for each port. INDEX {dot3adAggPortIndex} [Implementation] Same as the standard.#1	Y
29	dot3adAggPortStatsLACPDUsRx {dot3adAggPortStatsEntry 1}	R/O	[Standard] The number of valid LACPDUs received on the AggregationPort. [Implementation] Same as the standard.#1	Y
30	dot3adAggPortStatsMarkerPDUsRx {dot3adAggPortStatsEntry 2}	R/O	[Standard] The number of valid MarkerPDUs received on the AggregationPort. [Implementation] Same as the standard.#1	Y
31	dot3adAggPortStatsMarkerResponsePDUsRx {dot3adAggPortStatsEntry 3}	R/O	[Standard] The number of valid MarkerResponsePDUs received on the AggregationPort. [Implementation] Same as the standard.#1 Because the Switch sends no markers, there must be no valid marker responses; as a result, the value is fixed at 0.	Y

#	Object identifier	Access	Implementation specifications	Implemented
32	dot3adAggPortStatsUnknownRx {dot3adAggPortStatsEntry 4}	R/O	[Standard] The number of frames received that were either carried as a Slow Protocols Ethernet type value (88-09) but contained an unknown PDU, or were addressed to the Slow Protocols group MAC address (01-80-C2-00-00-02) but not carried as a Slow Protocols Ethernet type. [Implementation] Of the packets addressed to the MAC address 0180C2000002, those that correspond to either of the following:#1 ● The packet's Ethernet type is other than 0x8809, or it has a tag. ● The packet's slow protocol type is between 4 and 10.	Y
33	dot3adAggPortStatsIllegalRx {dot3adAggPortStatsEntry 5}	R/O	[Standard] The number of frames received that were either carried as a Slow Protocols Ethernet type value (88-09) but contained an unknown PDU, or contained an invalid Protocol Subtype value. [Implementation] Same as the standard.#1	Y
34	dot3adAggPortStatsLACPDUsTx {dot3adAggPortStatsEntry 6}	R/O	[Standard] The number of LACPDUs sent on the AggregationPort. [Implementation] Same as the standard.#1	Y
35	dot3adAggPortStatsMarkerPDUsTx {dot3adAggPortStatsEntry 7}	R/O	[Standard] The number of MarkerPDUs sent on the AggregationPort. [Implementation] Same as the standard.#1 Because this Switch sends no markers, the value is effectively fixed at 0.	Y
36	dot3adAggPortStatsMarkerResponsePDUsTx {dot3adAggPortStatsEntry 8}	R/O	[Standard] The number of MarkerResponsePDUs sent on the AggregationPort. [Implementation] Same as the standard.#1	Y
37	dot3adAggPortDebugTable {dot3adAggPort 3}	NA	[Standard] A table of link aggregation debug information for all ports. [Implementation] Same as the standard.#2	Y
38	dot3adAggPortDebugEntry {dot3adAggPortDebugTable 1}	NA	[Standard] A list of debug parameters for a port. INDEX {dot3adAggPortIndex} [Implementation] Same as the standard.#2	Y
39	dot3adAggPortDebugRxState {dot3adAggPortDebugEntry 1}	R/O	[Standard] The state of the Receive state machine for the AggregationPort: ● currentRx (1) ● expired (2) ● defaulted (3) ● initialize (4) ● lacpDisabled (5) ● portDisabled (6) [Implementation] Same as the standard.#2	Y

#	Object identifier	Access	Implementation specifications	Implemented
40	dot3adAggPortDebugLastRxTime {dot3adAggPortDebugEntry 2}	R/O	[Standard] The aTimeSinceSystemReset value for when the AggregationPort received the last LACPDU. [Implementation] Same as the standard.#2	Y
41	dot3adAggPortDebugMuxState {dot3adAggPortDebugEntry 3}	R/O	[Standard] The state of the Mux state machine for the AggregationPort: ● detached (1) ● waiting (2) ● attached (3) ● collecting (4) ● distributing (5) ● collectingDistributing (6) [Implementation] Same as the standard.#2	Y
42	dot3adAggPortDebugMuxReason {dot3adAggPortDebugEntry 4}	R/O	[Standard] The reason for the most recent state change of the Mux state machine. [Implementation] Empty string at all times.#2	M
43	dot3adAggPortDebugActorChurnState {dot3adAggPortDebugEntry 5}	R/O	[Standard] The state of the ActorChurnDetection state machine for the AggregationPort. [Implementation] Same as the standard.#2	Y
44	dot3adAggPortDebugPartnerChurnState {dot3adAggPortDebugEntry 6}	R/O	[Standard] The state of the PartnerChurnDetection state machine for the AggregationPort. [Implementation] Same as the standard.#2	Y
45	dot3adAggPortDebugActorChurnCount {dot3adAggPortDebugEntry 7}	R/O	[Standard] The number of times the state of the ActorChurn state machine changed to ACTOR_CHURN. [Implementation] Same as the standard.#2	Y
46	dot3adAggPortDebugPartnerChurnCount {dot3adAggPortDebugEntry 8}	R/O	[Standard] The number of times the state of the PartnerChurn state machine changed to PARTNER_CHURN. [Implementation] Same as the standard.#2	Y
47	dot3adAggPortDebugActorSyncTransitionCount {dot3adAggPortDebugEntry 9}	R/O	[Standard] The number of times the state of the Mux state machine for the Actor changed to IN_SYNC. [Implementation] Same as the standard.#2	Y
48	dot3adAggPortDebugPartnerSyncTransitionCount {dot3adAggPortDebugEntry 10}	R/O	[Standard] The number of times the state of the Mux state machine for the Partner changed to IN_SYNC. [Implementation] Same as the standard#2	Y
49	dot3adAggPortDebugActorChangeCount {dot3adAggPortDebugEntry 11}	R/O	[Standard] The number of times the Actor's perception of the LAG ID for the AggregationPort changed. [Implementation] Same as the standard.#2	Y

#	Object identifier	Access	Implementation specifications	Implemented
50	dot3adAggPortDebugPartnerChangeCount {dot3adAggPortDebugEntry 12}	R/O	[Standard] The number of times the Partner's perception of the LAG ID for the AggregationPort changed. [Implementation] Same as the standard.#2	Y

#1:

The value of dot3adAggPortStatsTable of any static port is void.

#2:

The value of dot3adAggPortDebugTable of any static port is void.

2.14.3 dot3adTablesLastChanged group

(1) ID

```
member-body   OBJECT IDENTIFIER ::= {iso 2}
us            OBJECT IDENTIFIER ::= {member-body 840}
ieee802dot3   OBJECT IDENTIFIER ::= {us 10006}
snmpmibs      OBJECT IDENTIFIER ::= {ieee802dot3 300}
lagMIB        OBJECT IDENTIFIER ::= {snmpmibs 43}
lagMIBObjects OBJECT IDENTIFIER ::= {lagMIB 1}

dot3adTablesLastChanged OBJECT IDENTIFIER ::= {lagMIBObjects 3}
Object ID value: 1.2.840.10006.300.43.1.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the dot3adTablesLastChanged group.

Table 2-32 dot3adTablesLastChanged group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot3adTablesLastChanged { lagMIBObjects 3 }	R/O	[Standard] The time when dot3adAggTable, dot3adAggPortListTable, or dot3adAggPortTable was last changed. [Implementation] Same as the standard.	Y

2.15 IEEE 802.1X MIB group

(1) ID

```
std             OBJECT IDENTIFIER ::= {iso 0}
iso8802         OBJECT IDENTIFIER ::= {std 8802}
ieee802dot1     OBJECT IDENTIFIER ::= {iso8802 1}
ieee802dot1mibs OBJECT IDENTIFIER ::= {ieee802dot1 1}
ieee8021paeMIB  OBJECT IDENTIFIER ::= {ieee802dot1mibs 1}
paeMIBObjects   OBJECT IDENTIFIER ::= {ieee8021paeMIB 1}

dot1xPaeSystem OBJECT IDENTIFIER ::= {paeMIBObjects 1}
Object ID value: 1.0.8802.1.1.1.1.1

dot1xPaeAuthenticator OBJECT IDENTIFIER ::= {paeMIBObjects 2}
Object ID value: 1.0.8802.1.1.1.1.2

dot1xPaeSupplicant OBJECT IDENTIFIER ::= {paeMIBObjects 3}
Object ID value: 1.0.8802.1.1.1.1.3

dot1xPaeConformance OBJECT IDENTIFIER ::= {ieee8021paeMIB 2}
dot1xPaeGroups      OBJECT IDENTIFIER ::= {dot1xPaeConformance 1}
Object ID value: 1.0.8802.1.1.1.2.1

dot1xPaeCompliances OBJECT IDENTIFIER ::= {dot1xPaeConformance 2}
Object ID value: 1.0.8802.1.1.1.2.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the IEEE 802.1X MIB group.

Table 2-33 IEEE 802.1X MIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	dot1xPaeSystemAuthControl {dot1xPaeSystem 1}	R/NW	[Standard] The administratively enabled or disabled state of the PAE (Port Access Entity) for the entire switch: INTEGER {enabled (1), disabled (2)} [Implementation] Same as the standard.	Y
2	dot1xPaePortTable {dot1xPaeSystem 2}	NA	[Standard] A table of system-level information for each PAE port. [Implementation] Same as the standard.	Y
3	dot1xPaePortEntry {dot1xPaePortTable 1}	NA	[Standard] A list of port-specific information. INDEX {dot1xPaePortNumber} [Implementation] Same as the standard.	Y
4	dot1xPaePortNumber {dot1xPaePortEntry 1}	NA	[Standard] A PAE port number. This number is used as an index to identify the table. [Implementation] Same as the standard. ifIndex is added to one of the following interfaces: ● Ethernet physical port ● VLAN group ● Link aggregation group	Y

#	Object identifier	Access	Implementation specifications	Implemented
5	dot1xPaePortProtocolVersion {dot1xPaePortEntry 2}	R/O	[Standard] The protocol version. [Implementation] Fixed value of 0x01.	Y
6	dot1xPaePortCapabilities {dot1xPaePortEntry 3}	R/O	[Standard] The PAE capability supported by the port: BITS {dot1xPaePortAuthCapable (0), dot1xPaePortSuppCapable (1)} [Implementation] Fixed value of dot1xPaePortAuthCapable (0).	Y
7	dot1xPaePortInitialize {dot1xPaePortEntry 4}	R/NW	[Standard] Initialization control for the port. When this attribute is set to TRUE, the port is initialized. The attribute returns to FALSE after initialization is complete. [Implementation] Same as the standard.	Y
8	dot1xPaePortReauthenticate {dot1xPaePortEntry 5}	R/NW	[Standard] Re-authentication control for the port. When this attribute is set to TRUE, the Authenticator PAE state machine for the port re-authenticates the Supplicant. There is no effect when this attribute is set to FALSE. This attribute returns to FALSE whenever read. [Implementation] Same as the standard.	Y
9	dot1xAuthConfigTable {dot1xPaeAuthenticator 1}	NA	[Standard] A table of configuration objects for the Authenticator PAE associated with each port. The table contains a list of ports that can be authenticated for access. [Implementation] Same as the standard.	Y
10	dot1xAuthConfigEntry {dot1xAuthConfigTable 1}	NA	[Standard] A list of configuration information for the Authenticator PAE. INDEX {dot1xPaePortNumber} [Implementation] Same as the standard.	Y
11	dot1xAuthPaeState {dot1xAuthConfigEntry 1}	R/O	[Standard] The current value of the Authenticator PAE state machine: ● initialize (1) ● disconnected (2) ● connecting (3) ● authenticating (4) ● authenticated (5) ● aborting (6) ● held (7) ● forceAuth (8) ● forceUnauth (9) [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
12	dot1xAuthBackendAuthState {dot1xAuthConfigEntry 2}	R/O	[Standard] The current value of the back-end authentication state machine: ● request (1) ● response (2) ● success (3) ● fail (4) ● timeout (5) ● idle (6) ● initialize (7) [Implementation] Same as the standard.	Y
13	dot1xAuthAdminControlledDirections {dot1xAuthConfigEntry 3}	R/NW	[Standard] The current value of the administratively controlled directions parameter for the port. [Implementation] Fixed value of both (0).	Y
14	dot1xAuthOperControlledDirections {dot1xAuthConfigEntry 4}	R/O	[Standard] The current value of the operationally controlled directions parameter for the port. [Implementation] Fixed value of both (0).	Y
15	dot1xAuthAuthControlledPortStatus {dot1xAuthConfigEntry 5}	R/O	[Standard] The current value of the control port state parameter for the port. [Implementation] Same as the standard.	Y
16	dot1xAuthAuthControlledPortControl {dot1xAuthConfigEntry 6}	R/NW	[Standard] The current value of the control port control parameter for the port. [Implementation] Same as the standard.	Y
17	dot1xAuthQuietPeriod {dot1xAuthConfigEntry 7}	R/NW	[Standard] The current quietPeriod constant value (in seconds) used by the Authenticator PAE state machine. DEFVAL{60} [Implementation] Same as the standard (0-65535).	Y
18	dot1xAuthTxPeriod {dot1xAuthConfigEntry 8}	R/NW	[Standard] The current txPeriod constant value (in seconds) used by the Authenticator PAE state machine. DEFVAL{30} [Implementation] Same as the standard (1-65535).	Y
19	dot1xAuthSuppTimeout {dot1xAuthConfigEntry 9}	R/NW	[Standard] The current suppTimeout constant value (in seconds) used by the back-end Authentication state machine. DEFVAL{30} [Implementation] Same as the standard (1-65535).	Y
20	dot1xAuthServerTimeout {dot1xAuthConfigEntry 10}	R/NW	[Standard] The current serverTimeout constant value (in seconds) used by the back-end Authentication state machine. DEFVAL{30} [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
21	dot1xAuthMaxReq {dot1xAuthConfigEntry 11}	R/NW	[Standard] The current maxReq constant value used by the back-end Authentication state machine. DEFVAL{2} [Implementation] Same as the standard (1-10).	Y
22	dot1xAuthReAuthPeriod {dot1xAuthConfigEntry 12}	R/NW	[Standard] The current reAuthperiod constant value (in seconds) used by the re-authentication timer state machine. DEFVAL{3600} [Implementation] 0 or from 1-65535. The default value is 3600. A value of 0 indicates that the Switch does not autonomously send an EAPOL-Request/Identity for re-authentication.	Y
23	dot1xAuthReAuthEnabled {dot1xAuthConfigEntry 13}	R/NW	[Standard] The enable-or-disable control used by the re-authentication timer state machine. DEFVAL{false (2)} [Implementation] Same as the standard.	Y
24	dot1xAuthKeyTxEnabled {dot1xAuthConfigEntry 14}	R/NW	[Standard] The current keyTransmissionEnabled constant value used by the Authenticator PAE state machine. [Implementation] Fixed value of false (2).	Y
25	dot1xAuthStatsTable {dot1xPaeAuthenticator 2}	NA	[Standard] A table of statistics objects for the Authenticator PAE associated with each port. The table contains a list of ports that can be authenticated for access. [Implementation] Same as the standard.	Y
26	dot1xAuthStatsEntry {dot1xAuthStatsTable 1}	NA	[Standard] Statistics for an Authenticator PAE. INDEX {dot1xPaePortNumber} [Implementation] Same as the standard.	Y
27	dot1xAuthEapolFramesRx {dot1xAuthStatsEntry 1}	R/O	[Standard] The number of all the valid EAPOL frames of each type received by the Authenticator. [Implementation] Same as the standard.	Y
28	dot1xAuthEapolFramesTx {dot1xAuthStatsEntry 2}	R/O	[Standard] The number of all the EAPOL frames of each type sent by the Authenticator. [Implementation] Same as the standard.	Y
29	dot1xAuthEapolStartFramesRx {dot1xAuthStatsEntry 3}	R/O	[Standard] The number of EAPOL Start frames received by the Authenticator. [Implementation] Same as the standard.	Y
30	dot1xAuthEapolLogoffFramesRx {dot1xAuthStatsEntry 4}	R/O	[Standard] The number of EAPOL Logoff frames received by the Authenticator. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
31	dot1xAuthEapolRespIdFramesRx {dot1xAuthStatsEntry 5}	R/O	[Standard] The number of EAP Response/Identity frames received by the Authenticator. [Implementation] Same as the standard.	Y
32	dot1xAuthEapolRespFramesRx {dot1xAuthStatsEntry 6}	R/O	[Standard] The number of EAP Response frames received by the Authenticator, excluding the EAP Response/Identity frames. [Implementation] Same as the standard.	Y
33	dot1xAuthEapolReqIdFramesTx {dot1xAuthStatsEntry 7}	R/O	[Standard] The number of EAP Request/Identity frames sent by the Authenticator. [Implementation] Same as the standard.	Y
34	dot1xAuthEapolReqFramesTx {dot1xAuthStatsEntry 8}	R/O	[Standard] The number of EAP Request frames sent by the Authenticator, excluding the EAP Request/Identity frames. [Implementation] Same as the standard.	Y
35	dot1xAuthInvalidEapolFramesRx {dot1xAuthStatsEntry 9}	R/O	[Standard] The number of EAPOL frames that were received by the Authenticator, but whose frame type was not recognized. [Implementation] Same as the standard.	Y
36	dot1xAuthEapLengthErrorFramesRx {dot1xAuthStatsEntry 10}	R/O	[Standard] The number of EAPOL frames that were received by the Authenticator, but whose Packet Body Length was invalid. [Implementation] Same as the standard.	Y
37	dot1xAuthLastEapolFrameVersion {dot1xAuthStatsEntry 11}	R/O	[Standard] The protocol version number of the last EAPOL frame received by the Authenticator. [Implementation] Same as the standard.	Y
38	dot1xAuthLastEapolFrameSource {dot1xAuthStatsEntry 12}	R/O	[Standard] The source MACAddress of the last EAPOL frame received by the Authenticator. [Implementation] Same as the standard.	Y
39	dot1xAuthDiagTable {dot1xPaeAuthenticator 3}	NA	[Standard] A table of diagnostic objects for the Authenticator PAE associated with each port. The table contains a list of ports that can be authenticated for access. [Implementation] Same as the standard.	Y
40	dot1xAuthDiagEntry {dot1xAuthDiagTable 1}	NA	[Standard] A list of diagnostic information about an Authenticator PAE INDEX {dot1xPaePortNumber} [Implementation] Same as the standard.	Y
41	dot1xAuthEntersConnecting {dot1xAuthDiagEntry 1}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed to CONNECTING. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
42	dot1xAuthEapLogoffsWhileConnecting {dot1xAuthDiagEntry 2}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from CONNECTING to DISCONNECTED because an EAPOL Logoff message was received. [Implementation] Same as the standard.	Y
43	dot1xAuthEntersAuthenticating {dot1xAuthDiagEntry 3}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from CONNECTING to AUTHENTICATING because an EAP Response/Identity message was received from a Supplicant. [Implementation] Same as the standard.	Y
44	dot1xAuthAuthSuccessWhileAuthenticating {dot1xAuthDiagEntry 4}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to AUTHENTICATED because the back-end authentication state machine indicated the Supplicant authentication was successful (authSuccess = TRUE). [Implementation] Same as the standard.	Y
45	dot1xAuthAuthTimeoutsWhileAuthenticating {dot1xAuthDiagEntry 5}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to ABORTING because the back-end authentication state machine indicated an authentication timeout (authTimeout = TRUE). [Implementation] Same as the standard.	Y
46	dot1xAuthAuthFailWhileAuthenticating {dot1xAuthDiagEntry 6}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to HELD because the back-end authentication state machine indicated there was an authentication failure (authFail = TRUE). [Implementation] Same as the standard.	Y
47	dot1xAuthAuthReauthsWhileAuthenticating {dot1xAuthDiagEntry 7}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to ABORTING because there was a re-authentication request (reAuthenticate = TRUE). [Implementation] Same as the standard.	Y
48	dot1xAuthAuthEapStartsWhileAuthenticating {dot1xAuthDiagEntry 8}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to ABORTING because an EAPOL Start message was received from a Supplicant. [Implementation] Same as the standard.	Y
49	dot1xAuthAuthEapLogoffWhileAuthenticating {dot1xAuthDiagEntry 9}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATING to ABORTING because an EAPOL Logoff message was received from a Supplicant. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
50	dot1xAuthAuthReauthsWhileAuthenticated {dot1xAuthDiagEntry 10}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATED to CONNECTING because there was a re-authentication request (reAuthenticate = TRUE). [Implementation] Same as the standard.	Y
51	dot1xAuthAuthEapStartsWhileAuthenticated {dot1xAuthDiagEntry 11}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATED to CONNECTING because an EAPOL Start message was received from a Supplicant. [Implementation] Same as the standard.	Y
52	dot1xAuthAuthEapLogoffWhileAuthenticated {dot1xAuthDiagEntry 12}	R/O	[Standard] The number of times the state of the Authenticator PAE state machine changed from AUTHENTICATED to DISCONNECTED because an EAPOL Logoff message was received from a Supplicant. [Implementation] Same as the standard.	Y
53	dot1xAuthBackendResponses {dot1xAuthDiagEntry 13}	R/O	[Standard] The number of times the back-end authentication state machine sent the first Access Request packet to the authentication server (by executing sendRespToServer when in a RESPONSE state). [Implementation] Same as the standard.	Y
54	dot1xAuthBackendAccessChallenges {dot1xAuthDiagEntry 14}	R/O	[Standard] The number of times the back-end authentication state machine received the first Access Challenge packet from the authentication server (that is, aReq was set to TRUE, causing the RESPONSE state to end). [Implementation] Same as the standard.	Y
55	dot1xAuthBackendOtherRequestsToSupplicant {dot1xAuthDiagEntry 15}	R/O	[Standard] The number of times the back-end authentication state machine sent an EAP Request (other than an Identity, Notification, Failure, or Success message) to a Supplicant (by executing txReq when in a REQUEST state). This implies that the Authenticator selects an EAP method. [Implementation] Same as the standard.	Y
56	dot1xAuthBackendNonNakResponsesFromSupplicant {dot1xAuthDiagEntry 16}	R/O	[Standard] The number of times the back-end authentication state machine received a response to the first EAP Request or a non-EAP NAK response from a Supplicant (that is, rxResp was set to TRUE, causing the state of the back-end authentication state machine to change from REQUEST to RESPONSE). (The response is not EAP NAK.) This implies that the Supplicant can respond to the EAP method selected by the Authenticator. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
57	dot1xAuthBackendAuthSuccesses {dot1xAuthDiagEntry 17}	R/O	[Standard] The number of times the back-end authentication state machine received an EAP Success message from the authentication server (that is, aSuccess was set to TRUE and the state of the back-end authentication state machine changed from RESPONSE to SUCCESS). This implies that the Supplicant was authenticated by the authentication server. [Implementation] Same as the standard.	Y
58	dot1xAuthBackendAuthFails {dot1xAuthDiagEntry 18}	R/O	[Standard] The number of times the back-end authentication state machine received an EAP Failure message from the authentication server (that is, aFail was set to TRUE and the state of the back-end authentication state machine changed from RESPONSE to FAIL). This implies that the Supplicant was not authenticated by the authentication server. [Implementation] Same as the standard.	Y
59	dot1xAuthSessionStatsTable {dot1xPaeAuthenticator 4}	NA	[Standard] A table of session statistics objects for the Authenticator PAE associated with each port. The table contains a list of ports that can be authenticated for access. [Implementation] Same as the standard.	Y
60	dot1xAuthSessionStatsEntry {dot1xAuthSessionStatsTable 1}	NA	[Standard] A list of session statistics for an Authenticator PAE. A set of counts collected for a session that is currently in progress, or a set of the last counts collected for the last valid session of a currently inactive port. INDEX {dot1xPaePortNumber} [Implementation] Same as the standard.	Y
61	dot1xAuthSessionOctetsRx {dot1xAuthSessionStatsEntry 1}	R/O	[Standard] The number of octets in the user data frames received on the port during the session. [Implementation] Fixed value of 0.	Y
62	dot1xAuthSessionOctetsTx {dot1xAuthSessionStatsEntry 2}	R/O	[Standard] The number of octets in the user data frames sent on the port during the session. [Implementation] Fixed value of 0.	Y
63	dot1xAuthSessionFramesRx {dot1xAuthSessionStatsEntry 3}	R/O	[Standard] The number of user data frames received on the port during the session. [Implementation] Fixed value of 0.	Y
64	dot1xAuthSessionFramesTx {dot1xAuthSessionStatsEntry 4}	R/O	[Standard] The number of user data frames sent on the port during the session. [Implementation] Fixed value of 0.	Y
65	dot1xAuthSessionId {dot1xAuthSessionStatsEntry 5}	R/O	[Standard] The unique identifier of the session. The identifier is a character string made up of three or more displayable ASCII characters. [Implementation] Fixed value of UnInitialized.	M

#	Object identifier	Access	Implementation specifications	Implemented
66	dot1xAuthSessionAuthenticMethod {dot1xAuthSessionStatsEntry 6}	R/O	[Standard] The authentication type used to establish the session: INTEGER {remoteAuthServer (1), localAuthServer (2)} [Implementation] Same as the standard. Fixed value of `remoteAuthServer` (1).	Y
67	dot1xAuthSessionTime {dot1xAuthSessionStatsEntry 7}	R/O	[Standard] The session hold time (in seconds). [Implementation] Same as the standard.	Y
68	dot1xAuthSessionTerminateCause {dot1xAuthSessionStatsEntry 8}	R/O	[Standard] The reason the session ended: ● supplicantLogoff (1) ● portFailure (2) ● supplicantRestart (3) ● reauthFailed (4) ● authControlForceUnauth (5) ● portReInit (6) ● portAdminDisabled (7) ● notTerminatedYet (999) [Implementation] Same as the standard. Fixed value of notTerminatedYet (999).	M
69	dot1xAuthSessionUserName {dot1xAuthSessionStatsEntry 9}	R/O	[Standard] A user name that identifies the Supplicant PAE. [Implementation] Fixed value of `UnInitialized`.	M
70	dot1xSuppConfigTable {dot1xPaeSupplicant 1}	NA	[Standard] A table of configuration objects for the Supplicant PAE associated with each port. The table contains a list of ports that can be authenticated for access by a remote system. [Implementation] Not implemented.	N
71	dot1xSuppConfigEntry {dot1xSuppConfigTable 1}	NA	[Standard] A list of configuration information about a Supplicant PAE. INDEX {dot1xPaePortNumber} [Implementation] Not implemented.	N
72	dot1xSuppPaeState {dot1xSuppConfigEntry 1}	R/O	[Standard] The current state of the Supplicant PAE state machine: ● disconnected (1) ● logoff (2) ● connecting (3) ● authenticating (4), ● authenticated (5) ● acquired (6) ● held (7) [Implementation] Not implemented.	N
73	dot1xSuppHeldPeriod {dot1xSuppConfigEntry 2}	R/NW	[Standard] The current heldPeriod constant value (in seconds) used by the Supplicant PAE state machine. DEFVAL{60} [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
74	dot1xSuppAuthPeriod {dot1xSuppConfigEntry 3}	R/NW	[Standard] The current authPeriod constant value (in seconds) used by the Supplicant PAE state machine. DEFVAL{30} [Implementation] Not implemented.	N
75	dot1xSuppStartPeriod {dot1xSuppConfigEntry 4}	R/NW	[Standard] The current startPeriod constant value (in seconds) used by the Supplicant PAE state machine. DEFVAL{30} [Implementation] Not implemented.	N
76	dot1xSuppMaxStart {dot1xSuppConfigEntry 5}	R/NW	[Standard] The current maxStart constant value (in seconds) used by the Supplicant PAE state machine. DEFVAL{3} [Implementation] Not implemented.	N
77	dot1xSuppStatsTable {dot1xPaeSupplicant 2}	NA	[Standard] A table of statistics objects for the Supplicant PAE associated with each port. The table contains a list of ports that can be authenticated for access by a remote system. [Implementation] Not implemented.	N
78	dot1xSuppStatsEntry {dot1xSuppStatsTable 1}	NA	[Standard] A list of statistics about a Supplicant PAE. INDEX {dot1xPaePortNumber} [Implementation] Not implemented.	N
79	dot1xSuppEapolFramesRx {dot1xSuppStatsEntry 1}	R/O	[Standard] The number of all the EAPOL frames of each type received by the Supplicant. [Implementation] Not implemented.	N
80	dot1xSuppEapolFramesTx {dot1xSuppStatsEntry 2}	R/O	[Standard] The number of all the EAPOL frames of all types sent by the Supplicant. [Implementation] Not implemented.	N
81	dot1xSuppEapolStartFramesTx {dot1xSuppStatsEntry 3}	R/O	[Standard] The number of EAPOL Start frames sent by the Supplicant. [Implementation] Not implemented.	N
82	dot1xSuppEapolLogoffFramesTx {dot1xSuppStatsEntry 4}	R/O	[Standard] The number of EAPOL Logoff frames sent by the Supplicant. [Implementation] Not implemented.	N
83	dot1xSuppEapolRespIdFramesTx {dot1xSuppStatsEntry 5}	R/O	[Standard] The number of EAP Response/Identity frames sent by the Supplicant. [Implementation] Not implemented.	N
84	dot1xSuppEapolRespFramesTx {dot1xSuppStatsEntry 6}	R/O	[Standard] The number of valid EAP Response frames sent by the Supplicant (excluding Response/Identity frames). [Implementation] Not implemented.	N

#	Object identifier	Access	Implementation specifications	Implemented
85	dot1xSuppEapolReqIdFramesRx {dot1xSuppStatsEntry 7}	R/O	[Standard] The number of EAP Request/Identity frames received by the Supplicant. [Implementation] Not implemented.	N
86	dot1xSuppEapolReqFramesRx {dot1xSuppStatsEntry 8}	R/O	[Standard] The number of EAP Request frames received by the Supplicant (excluding Request/Identity frames). [Implementation] Not implemented.	N
87	dot1xSuppInvalidEapolFramesRx {dot1xSuppStatsEntry 9}	R/O	[Standard] The number of EAPOL frames that were received by the Supplicant, but whose frame type was not recognized. [Implementation] Not implemented.	N
88	dot1xSuppEapLengthErrorFramesRx {dot1xSuppStatsEntry 10}	R/O	[Standard] The number of EAPOL frames that were received by the Supplicant, but whose Packet Body Length was invalid. [Implementation] Not implemented.	N
89	dot1xSuppLastEapolFrameVersion {dot1xSuppStatsEntry 11}	R/O	[Standard] The protocol version number of the last EAPOL frame received by the Supplicant. [Implementation] Not implemented.	N
90	dot1xSuppLastEapolFrameSource {dot1xSuppStatsEntry 12}	R/O	[Standard] The source MAC address of the last EAPOL frame received by the Supplicant. [Implementation] Not implemented.	N

2.16 snmpModules group

2.16.1 snmpFrameworkMIB group (SNMP FRAMEWORK MIB)

The snmpFrameworkMIB group is described in the following document:

- RFC 3411 (December 2002)

(1) ID

```
snmpFrameworkMIB MODULE-IDENTITY ::= {snmpModules 10}

snmpFrameworkMIBObjects OBJECT IDENTIFIER ::= {snmpFrameworkMIB 2}
Object ID value: 1.3.6.1.6.3.10.2

snmpEngine              OBJECT IDENTIFIER ::= {snmpFrameworkMIBObjects 1}
Object ID value: 1.3.6.1.6.3.10.2.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpFrameworkMIB group.

Table 2-34 snmpFrameworkMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	snmpEngineID {snmpEngine 1}	R/O	[Standard] An ID for SNMP engine management. However, all 0, all 0xff, and null (0-byte length) are not permitted. [Implementation] Same as the standard. For details, see the configuration command snmp-server engineID local.	Y
2	snmpEngineBoots {snmpEngine 2}	R/O	[Standard] The number of times re-initialization has occurred since snmpEngineID was last set. [Implementation] Same as the standard. The SNMP agent in this Switch is also initialized when power saving sleep ends.	Y
3	snmpEngineTime {snmpEngine 3}	R/O	[Standard] The time that has elapsed (in seconds) since snmpEngineBoots was incremented. Note that when this exceeds the maximum value, it is reset to 0 and snmpEngineBoots is incremented. [Implementation] Same as the standard.	Y
4	snmpEngineMax MessageSize {snmpEngine 4}	R/O	[Standard] The maximum size of the message that can be sent or received by the SNMP engine. [Implementation] Same as the standard. Fixed value of 2048.	Y

2.16.2 snmpMPDMIB group (SNMP MPD MIB)

The snmpMPDMIB group is described in the following document:

- RFC 3412 (December 2002)

(1) ID

```
snmpMPDMIB MODULE-IDENTITY ::= {snmpModules 11}

snmpMPDMIBObjects OBJECT IDENTIFIER ::= {snmpMPDMIB 2}
Object ID value: 1.3.6.1.6.3.11.2

snmpMPDStats      OBJECT IDENTIFIER ::= {snmpMPDMIBObjects 1}
Object ID value: 1.3.6.1.6.3.11.2.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpMPDMIB group.

Table 2-35 snmpMPDMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	snmpUnknownSecurityModels {snmpMPDStats 1}	R/O	[Standard] The total number of packets received but discarded due to an unsupported securityModel. [Implementation] Same as the standard.	Y
2	snmpInvalidMsgs {snmpMPDStats 2}	R/O	[Standard] The total number of packets received but discarded due to an invalid message. [Implementation] Same as the standard.	Y
3	snmpUnknownPDUHandlers {snmpMPDStats 3}	R/O	[Standard] The total number of packets received but discarded because there was a PDU that could not be processed by an application. [Implementation] Same as the standard.	Y

2.16.3 snmpTargetMIB group (SNMP TARGET MIB)

The snmpTargetMIB group is described in the following document:

- RFC 3413 (December 2002)

(1) ID

```
snmpTargetMIB MODULE-IDENTITY ::= {snmpModules 12}

snmpTargetObjects OBJECT IDENTIFIER ::= {snmpTargetMIB 1}
Object ID value: 1.3.6.1.6.3.12.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpTargetMIB group.

Table 2-36 snmpTargetMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	snmpTargetSpinLock {snmpTargetObjects 1}	R/NW	[Standard] Used for locking if multiple managers request the modification of the table entries in the SNMP-TARGET-MIB module. [Implementation] Undefined value.	M

#	Object identifier	Access	Implementation specifications	Implemented
2	snmpTargetAddrTable {snmpTargetObjects 2}	NA	[Standard] A table of transmission addresses used in SNMP message generation. [Implementation] Same as the standard. Provides information on SNMPv3 registered through the configuration command `snmp-server host`.	Y
3	snmpTargetAddrEntry {snmpTargetAddrTable 1}	NA	[Standard] An entry for a transmission address used in SNMP message generation. INDEX { IMPLIED snmpTargetAddrName } [Implementation] Same as the standard.	Y
4	snmpTargetAddrName {snmpTargetAddrEntry 1}	NA	[Standard] The name of the snmpTargetAddEntry. [Implementation] Same as the standard. Corresponds to *manager-address* for the configuration command `snmp-server host`.	Y
5	snmpTargetAddrTDomain {snmpTargetAddrEntry 2}	R/NC	[Standard] The transmission type of the snmpTargetAddrTAddress object address. [Implementation] Same as the standard.	Y
6	snmpTargetAddrTAddress {snmpTargetAddrEntry 3}	R/NC	[Standard] The transmission address. The format of this address is indicated by snmpTargetAddrTDomain. [Implementation] Same as the standard. IP address and UDP port number of the SNMP manager.	Y
7	snmpTargetAddrTimeout {snmpTargetAddrEntry 4}	R/NC	[Standard] The timeout value (in tens of milliseconds) for communication with the transmission address defined by this entry. The default value is `1500`. [Implementation] Same as the standard. Fixed value of `0`.	Y
8	snmpTargetAddrRetryCount {snmpTargetAddrEntry 5}	R/NC	[Standard] The default number of retries if there is no response to a sent message. The default value is `3`. [Implementation] Same as the standard. Fixed value of `0`.	Y
9	snmpTargetAddrTagList {snmpTargetAddrEntry 6}	R/NC	[Standard] A snmpNotifyTag list. The default value is blank. [Implementation] Same as the standard. Fixed value of `TRAP`.	Y
10	snmpTargetAddrParams {snmpTargetAddrEntry 7}	R/NC	[Standard] A snmpTargetParamsTable entry. [Implementation] Same as the standard. Has the same value as `snmpTargetAddrName`.	Y
11	snmpTargetAddrStorageType {snmpTargetAddrEntry 8}	R/NC	[Standard] The saving format of this entry. The default value is `nonVolatile`. [Implementation] Fixed value of readOnly (`5`).	Y

#	Object identifier	Access	Implementation specifications	Implemented
12	snmpTargetAddrRowStatus {snmpTargetAddrEntry 9}	R/NC	[Standard] Status of this entry: If a new entry is added, notReady (3) is set until snmpTargetAddrTDomain, snmpTargetAddrTAddress, and snmpTargetAddrParams are set. However, if this object is active (1), snmpTargetAddrTDomain and snmpTargetAddrTAddress cannot be changed. [Implementation] Same as the standard. Fixed value of active (1).	Y
13	snmpTargetParamsTable {snmpTargetObjects 3}	NA	[Standard] A table of SNMP target information used in SNMP message generation. [Implementation] Same as the standard. Provides information on SNMPv3 registered through the configuration command `snmp-server host`.	Y
14	snmpTargetParamsEntry {snmpTargetParamsTable 1}	NA	[Standard] An entry for the SNMP target information used in SNMP message generation. INDEX { IMPLIED snmpTargetParamsName } [Implementation] Same as the standard.	Y
15	snmpTargetParamsName {snmpTargetParamsEntry 1}	NA	[Standard] The name of snmpTargetParamsEntry. [Implementation] Same as the standard. Corresponds to *manager-address* for the configuration command `snmp-server host`.	Y
16	snmpTargetParamsMPModel {snmpTargetParamsEntry 2}	R/NC	[Standard] The message processing model used in SNMP message generation. Values from 0 to 255 are managed by IANA. ● 0: SNMPv1 ● 1: SNMPv2C ● 2: SNMPv2u, SNMPv2* ● 3: SNMPv3 256 or higher is enterprise-unique. [Implementation] Same as the standard. Fixed value of SNMPv3 (3).	Y
17	snmpTargetParamsSecurityModel {snmpTargetParamsEntry 3}	R/NC	[Standard] The security model for creating an SNMP message. Values from 1 to 255 are managed by IANA. ● 0: No particular model provided ● 1: SNMPv1 ● 2: SNMPv2C ● 3: User-Based Security Model (USM) 256 or higher is enterprise-unique. [Implementation] Same as the standard. Fixed value of USM (3).	Y
18	snmpTargetParamsSecurityName {snmpTargetParamsEntry 4}	R/NC	[Standard] The securityName indicating the method used in SNMP message generation. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
19	snmpTargetParamsSecurityLevel {snmpTargetParamsEntry 5}	R/NC	[Standard] The security level in SNMP message generation: ● noAuthNoPriv (1): No authentication, no privacy ● authNoPriv (2): Authentication provided, no privacy ● authPriv (3): Authentication and privacy provided [Implementation] Same as the standard.	Y
20	snmpTargetParamsStorageType {snmpTargetParamsEntry 6}	R/NC	[Standard] The saving format of this entry. [Implementation] Fixed value of readOnly (5).	Y
21	snmpTargetParamsRowStatus {snmpTargetParamsEntry 7}	R/NC	[Standard] Status of this entry: If a new entry is added, notReady (3) is set until snmpTargetParamsMPModel, snmpTargetParamsSecurityModel, snmpTargetParamsSecurityName, and snmpTargetParamsSecurityLevel are set. However, if this object is active (1), snmpTargetParamsMPModel, snmpTargetParamsSecurityModel, snmpTargetParamsSecurityName, and snmpTargetParamsSecurityLevel cannot be changed. [Implementation] Same as the standard. Fixed value of active (1).	Y
22	snmpUnavailableContexts {snmpTargetObjects 4}	R/O	[Standard] The total number of packets received but discarded due to an unusable context in the message. [Implementation] Same as the standard. Fixed value of 0.	Y
23	snmpUnknownContexts {snmpTargetObjects 5}	R/O	[Standard] The total number of packets received but discarded due to an unreadable context in the message. [Implementation] Same as the standard.	Y

2.16.4 snmpNotificationMIB group (SNMP NOTIFICATION MIB)

The snmpNotificationMIB group is described in the following document:

- RFC 3413 (December 2002)

(1) ID

```
snmpNotificationMIB MODULE-IDENTITY ::= {snmpModules 13}

snmpNotifyObjects OBJECT IDENTIFIER ::= {snmpNotificationMIB 1}
Object ID value: 1.3.6.1.6.3.13.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpNotificationMIB group.

Table 2-37 snmpNotificationMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	snmpNotifyTable {snmpNotifyObjects 1}	NA	[Standard] A table used to identify a managed target that receives a notification, and the type of the notification that is sent to the target. [Implementation] Same as the standard.	Y
2	snmpNotifyEntry {snmpNotifyTable 1}	NA	[Standard] An entry for a managed target that receives a notification, and the type of the notification that is sent to the target. INDEX { IMPLIED snmpNotifyName } [Implementation] Same as the standard.	Y
3	snmpNotifyName {snmpNotifyEntry 1}	NA	[Standard] The name of snmpNotifyEntry. [Implementation] Same as the standard. Have the same value as snmpNotifyTag.	Y
4	snmpNotifyTag {snmpNotifyEntry 2}	R/NC	[Standard] The tag value used to identify the snmpTargetAddrTable entry. The default value is blank. [Implementation] Same as the standard. Fixed value of TRAP.	Y
5	snmpNotifyType {snmpNotifyEntry 3}	R/NC	[Standard] The type of a notification. The default value is trap (1). ● trap (1) ● inform (2) [Implementation] Same as the standard. Fixed value of trap (1).	Y
6	snmpNotifyStorageType {snmpNotifyEntry 4}	R/NC	[Standard] The saving format of this entry. The default value is nonVolatile. [Implementation] Fixed value of readOnly (5).	Y
7	snmpNotifyRowStatus {snmpNotifyEntry 5}	R/NC	[Standard] Status of this entry. [Implementation] Same as the standard. Fixed value of active (1).	Y
8	snmpNotifyFilterProfileTable {snmpNotifyObjects 2}	NA	[Standard] A table used to associate a notification filter definition with a specific target parameter. [Implementation] Same as the standard.	Y
9	snmpNotifyFilterProfileEntry {snmpNotifyFilterProfileTable 1}	NA	[Starndard] An entry for a filter definition used in notification generation. INDEX { IMPLIED snmpTargetParamsName } [Implementation] Same as the standard.	Y
10	snmpNotifyFilterProfileName {snmpNotifyFilterProfileEntry 1}	R/NC	[Standard] The name of the filter definition. This is associated with snmpTargetParamsTable. [Implementation] Same as the standard. Has the same value as snmpTargetAddrName.	Y

#	Object identifier	Access	Implementation specifications	Implemented
11	snmpNotifyFilterProfileStorageType {snmpNotifyFilterProfileEntry 2}	R/NC	[Standard] The saving format of this entry. The default value is nonVolatile. [Implementation] Fixed value of readOnly (5).	Y
12	snmpNotifyFilterProfileRowStatus {snmpNotifyFilterProfileEntry 3}	R/NC	[Standard] Status of this entry: If a new entry is added, notReady (3) is set until snmpNofityFilterProfileName is set. [Implementation] Same as the standard. Fixed value of active (1).	Y
13	snmpNotifyFilterTable {snmpNotifyObjects 3}	NA	[Standard] A table of filter definitions used to determine whether a management target receives a notification. [Implementation] Same as the standard. Provides information on SNMPv3 registered through the configuration command snmp-server host.	Y
14	snmpNotifyFilterEntry {snmpNotifyFilterTable 1}	NA	[Standard] An entry for a filter definition used to determine whether a management target receives a notification. INDEX { snmpNotifyFilterProfileName, IMPLIED snmpNotifyFilterSubtree } [Implementation] Same as the standard.	Y
15	snmpNotifyFilterSubtree {snmpNotifyFilterEntry 1}	NA	[Standard] A MIB subtree, when combined with the corresponding instance of snmpNotifyFilterMask, defines a family of subtrees to be included or excluded by the filter profile. [Implementation] Same as the standard.	Y
16	snmpNotifyFilterMask {snmpNotifyFilterEntry 2}	R/NC	[Standard] A bit mask that, when combined with the corresponding instance of snmpNotifyFilterSubtree, defines a family of subtrees to be included or excluded by the filter profile. ● '1': Exact match ● '0': Wildcard If the object length is 0, this extended rule is padded with all 1s. The filter subtree family becomes a subtree that is uniquely identified by the corresponding instance of snmpNofityFilterSubtree. The default value is ''H. [Implementation] Same as the standard.	Y
17	snmpNotifyFilterType {snmpNotifyFilterEntry 3}	R/NC	[Standard] Indicates whether the filter subtree family defined by this entry is included in the filter. The default value is included. ● included (1) ● excluded (2) [Implementation] Same as the standard.	Y
18	snmpNotifyFilterStorageType {snmpNotifyFilterEntry 4}	R/NC	[Standard] The saving format of this entry. The default value is nonVolatile. [Implementation] Fixed value of readOnly (5).	Y

#	Object identifier	Access	Implementation specifications	Implemented
19	snmpNotifyFilterRowStatus {snmpNotifyFilterEntry 5}	R/NC	[Standard] Status of this entry. [Implementation] Same as the standard. Fixed value of active (1).	Y

2.16.5 snmpUsmMIB group (SNMP USER BASED SM MIB)

The snmpUsmMIB group is described in the following document:

- RFC 3414 (December 2002)

(1) ID

```
snmpUsmMIB MODULE-IDENTITY ::= {snmpModules 15}

usmMIBObjects OBJECT IDENTIFIER ::= {snmpUsmMIB 1}
Object ID value: 1.3.6.1.6.3.15.1
usmStats      OBJECT IDENTIFIER ::= {usmMIBObjects 1}
Object ID value: 1.3.6.1.6.3.15.1.1

usmUser       OBJECT IDENTIFIER ::= {usmMIBObjects 2}
Object ID value: 1.3.6.1.6.3.15.1.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpUsmMIB group.

Table 2-38 snmpUsmMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	usmStatsUnsupportedSecLevels {usmStats 1}	R/O	[Standard] The total number of packets received but discarded due to an invalid security level. [Implementation] Same as the standard.	Y
2	usmStatsNotInTimeWindows {usmStats 2}	R/O	[Standard] The total number of packets received but discarded because WindowTime was out of range. [Implementation] Same as the standard.	Y
3	usmStatsUnknownUserNames {usmStats 3}	R/O	[Standard] The total number of packets received but discarded due to an invalid user. [Implementation] Same as the standard.	Y
4	usmStatsUnknownEngineIDs {usmStats 4}	R/O	[Standard] The total number of packets received but discarded because an unknown snmpEngineID was referenced. [Implementation] Same as the standard.	Y
5	usmStatsWrongDigests {usmStats 5}	R/O	[Standard] The total number of packets received but discarded because they did not contain the expected digest value. [Implementation] Same as the standard.	Y
6	usmStatsDecryptionErrors {usmStats 6}	R/O	[Standard] The total number of packets received but discarded because decryption failed. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
7	usmUserSpinLock {usmUser 1}	R/NW	[Standard] A lock used for altering secrets in the usmUserTable. [Implementation] Undefined value.	M
8	usmUserTable {usmUser 2}	NA	[Standard] A user table configured in the LCD (Local Configuration Datastore) of the SNMP engine. [Implementation] Same as the standard. Provides information registered through the configuration command `snmp-server user`.	Y
9	usmUserEntry {usmUserTable 1}	NA	[Standard] An entry in the user table configured in the LCD (Local Configuration Datastore) of the SNMP engine. INDEX { usmUserEngineID, usmUserName } [Implementation] Same as the standard.	Y
10	usmUserEngineID {usmUserEntry 1}	NA	[Standard] An ID for SNMP engine management. [Implementation] Same as the standard. Has the same value as snmpEngineID.	Y
11	usmUserName {usmUserEntry 2}	NA	[Standard] Legible name indicating the user. This is the security ID on which USM depends. [Implementation] Same as the standard.	Y
12	usmUserSecurity Name {usmUserEntry 3}	R/O	[Standard] Legible name indicating a user for which the format does not depend on the security model. This name has the same value as usmUserName. [Implementation] Same as the standard.	Y
13	usmUserCloneFrom {usmUserEntry 4}	R/NC	[Standard] A pointer to a different entry that is used as a copy source to add a new entry. If this object is read, an object ID of `0.0` is returned. [Implementation] Object identifier `zeroDotZero`.	Y
14	usmUserAuthProtocol {usmUserEntry 5}	R/NC	[Standard] An authentication protocol of the SNMP engine indicated by usmUserEngineID. [Implementation] Same as the standard.	Y
15	usmUserAuthKeyChange {usmUserEntry 6}	R/NC	[Standard] An object that generates an authentication key for the SNMP engine indicated by usmUserEngineID. This is set if the requesting usmUserName is not the same as the usmUserName for this entry. If this object is read, a 0-length character string is returned. The default value is `''H`. [Implementation] An empty string.	Y

#	Object identifier	Access	Implementation specifications	Implemented
16	usmUserOwnAuthKeyChange {usmUserEntry 7}	R/NC	[Standard] An object that generates an authentication key for the SNMP engine indicated by usmUserEngineID. This is set if the requesting usmUserName is the same as the usmUserName for this entry. If this object is read, a 0-length character string is returned. The default value is ''H. [Implementation] An empty string.	Y
17	usmUserPrivProtocol {usmUserEntry 8}	R/NC	[Standard] Used by the privacy protocol of the SNMP engine indicated by usmUserEngineID. The default value is usmNoPrivProtocol. [Implementation] Same as the standard.	Y
18	usmUserPrivKeyChange {usmUserEntry 9}	R/NC	[Standard] An object that generates the encryption key indicated by the usmUserEngineID. This is set if the requesting usmUserName is not the same as the usmUserName for this entry. If this object is read, a 0-length character string is returned. The default value is ''H. [Implementation] An empty string.	Y
19	usmUserOwnPrivKeyChange {usmUserEntry 10}	R/NC	[Standard] An object that generates the encryption key indicated by the usmUserEngineID. This is set if the requesting usmUserName is the same as the usmUserName for this entry. If this object is read, a 0-length character string is returned. The default value is ''H. [Implementation] An empty string.	Y
20	usmUserPublic {usmUserEntry 11}	R/NC	[Standard] A value generated in the process of changing a user authentication or encryption key. This can be used later to deteremine whether the key change is valid. The default value is ''H. [Implementation] An empty string.	Y
21	usmUserStorageType {usmUserEntry 12}	R/NC	[Standard] The saving format of this entry. [Implementation] Fixed value of readOnly (5).	Y
22	usmUserStatus {usmUserEntry 13}	R/NC	[Standard] Status of this entry. [Implementation] Same as the standard. Fixed value of active (1).	Y

2.16.6 snmpVacmMIB group (SNMP VIEW BASED ACM MIB)

The snmpVacmMIB group is described in the following document:

- RFC 3415 (December 2002)

(1) ID

```
snmpVacmMIB MODULE-IDENTITY ::= {snmpModules 16}

vacmMIBObjects OBJECT IDENTIFIER ::= {snmpVacmMIB 1}
Object ID value: 1.3.6.1.6.3.16.1

vacmMIBViews   OBJECT IDENTIFIER ::= {vacmMIBObjects 5}
Object ID value: 1.3.6.1.6.3.16.1.5
```

(2) Implementation specifications

The following table shows the implementation specifications for the snmpVacmMIB group.

Table 2-39 snmpVacmMIB group implementation specifications

#	Object identifier	Access	Implementation specifications	Implemented
1	vacmContextTable {vacmMIBObjects 1}	NA	[Standard] A context table that can be used locally. [Implementation] Same as the standard. This device has only the default context.	Y
2	vacmContextEntry {vacmContextTable 1}	NA	[Standard] A context table entry that can be used locally. INDEX { vacmContextName } [Implementation] Same as the standard.	Y
3	vacmContextName {vacmContextEntry 1}	R/O	[Standard] Legible name indicating a particular context of a particular SNMP entity. An empty contextName indicates the default context. [Implementation] Same as the standard. An empty string.	Y
4	vacmSecurityToGroupTable {vacmMIBObjects 2}	NA	[Standard] A table used to define an access control policy for an operator group. [Implementation] Same as the standard. Provides information registered through the configuration command `snmp-server user`.	Y
5	vacmSecurityToGroupEntry {vacmSecurityToGroupTable 1}	NA	[Standard] An entry used to define an access control policy for an operator group. The entry indicates a groupName that consists of a securityModel and securityName. INDEX { vacmSecurityModel, vacmSecurityName } [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
6	vacmSecurityModel {vacmSecurityToGroup Entry 1}	NA	[Standard] A security model of vacmSecurityName referenced by this entry. A value of 0 cannot be specified. Values from 1 to 255 are managed by IANA. ● 0: No particular model provided. ● 1: SNMPv1 ● 2: SNMPv2C ● 3: User-Based Security Model (USM) 256 or higher is enterprise-unique. [Implementation] Same as the standard. Fixed value of USM (3).	Y
7	vacmSecurityName {vacmSecurityToGroup Entry 2}	NA	[Standard] The securityName of this entry. This entry is mapped to groupName. [Implementation] Same as the standard.	Y
8	vacmGroupName {vacmSecurityToGroup Entry 3}	R/NC	[Standard] The name of the group to which this entry belongs. [Implementation] Same as the standard.	Y
9	vacmSecurityToGroupS torageType {vacmSecurityToGroup Entry 4}	R/NC	[Standard] The saving format of this entry. The default value is nonVolatile. [Implementation] Fixed value of readOnly (5).	Y
10	vacmSecurityToGroupS tatus {vacmSecurityToGroup Entry 5}	R/NC	[Standard] Status of this entry: If a new entry is added, notReady (3) is set until vacmGroupName is set. [Implementation] Same as the standard. Fixed value of active (1).	Y
11	vacmAccessTable {vacmMIBObjects 4}	NA	[Standard] A table of access permissions for groups. [Implementation] Same as the standard. Provides information registered through the configuration command snmp-server group.	Y
12	vacmAccessEntry {vacmAccessTable 1}	NA	[Standard] An entry for access permissions for a group. INDEX { vacmGroupName, vacmAccessContextPrefix, vacmAccessSecurityModel, vacmAccessSecurityLevel } [Implementation] Same as the standard.	Y
13	vacmAccessContextPr efix {vacmAccessEntry 1}	NA	[Standard] A value that is compared with this entry to gain access permission. [Implementation] Same as the standard. An empty string denoting the default context.	Y

#	Object identifier	Access	Implementation specifications	Implemented
14	vacmAccessSecurityModel {vacmAccessEntry 2}	NA	[Standard] A securityModel needed to gain access permission for this entry. Values from 1 to 255 are managed by IANA. ● 0: No specific model ● 1: SNMPv1 ● 2: SNMPv2C ● 3: User-Based Security Model (USM) 256 or higher is enterprise-unique. [Implementation] Same as the standard. Fixed value of USM (3).	Y
15	vacmAccessSecurityLevel {vacmAccessEntry 3}	NA	[Standard] A security level needed to gain access permission for this entry. ● noAuthNoPriv (1): No authentication, no privacy ● authNoPriv (2): Authentication provided, no privacy ● authPriv (3): Authentication and privacy provided [Implementation] Same as the standard.	Y
16	vacmAccessContextMatch {vacmAccessEntry 4}	R/NC	[Standard] ● exact (1): Selects all line entries whose contextName perfectly matches vacmAccessContextPrefix. ● prefix (2): Selects all line entries whose contextName begins with vacmAccessContextPrefix. The default value is exact. [Implementation] Same as the standard. Fixed value of exact (1).	Y
17	vacmAccessReadViewName {vacmAccessEntry 5}	R/NC	[Standard] The vacmViewTreeFamilyViewName for a MIB view to which this entry authenticates read access. The default value is ''H. [Implementation] Same as the standard.	Y
18	vacmAccessWriteViewName {vacmAccessEntry 6}	R/NC	[Standard] The vacmViewTreeFamilyViewName for a MIB view to which this entry authenticates write access. The default value is ''H. [Implementation] Same as the standard.	Y
19	vacmAccessNotifyViewName {vacmAccessEntry 7}	R/NC	[Standard] The vacmViewTreeFamilyViewName for a MIB view to which this entry authenticates notifications access. The default value is ''H. [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
20	vacmAccessStorageType {vacmAccessEntry 8}	R/NC	[Standard] The saving format of this entry. The default value is `nonVolatile`. [Implementation] Fixed value of readOnly (5).	Y
21	vacmAccessStatus {vacmAccessEntry 9}	R/NC	[Standard] Status of this entry. [Implementation] Same as the standard. Fixed value of active (1).	Y
22	vacmViewSpinLock {vacmMIBViews 1}	R/NW	[Standard] An advisory lock used to allow an application for a cooperating SNMP command generator to coordinate its SET operation in view creation or modification. This is an advisory lock, and its use is not enforced. [Implementation] Undefined value.	M
23	vacmViewTreeFamilyTable {vacmMIBViews 2}	NA	[Standard] A locally held table of information about the subtree families for a MIB view. All the view subtrees (including those to be included or excluded) are defined in this table. [Implementation] Same as the standard. Provides information registered through the configuration command `snmp-server view`.	Y
24	vacmViewTreeFamilyEntry {vacmViewTreeFamilyTable 1}	NA	[Standard] A locally held entry for information about a subtree family for a MIB view. INDEX { vacmViewTreeFamilyViewName, vacmViewTreeFamilySubtree } [Implementation] Same as the standard.	Y
25	vacmViewTreeFamilyViewName {vacmViewTreeFamilyEntry 1}	NA	[Standard] Legible name of a view subtree family. [Implementation] Same as the standard.	Y
26	vacmViewTreeFamilySubtree {vacmViewTreeFamilyEntry 2}	NA	[Standard] A MIB subtree that defines the view subtree family. [Implementation] Same as the standard.	Y
27	vacmViewTreeFamilyMask {vacmViewTreeFamilyEntry 3}	R/NC	[Standard] A mask value for a vacmViewTreeFamilySubtree: ● 1: An exact match must occur. ● 0: Indicates a wildcard. If the length of this object is 0, a mask of all 1s is used. [Implementation] Same as the standard.	Y
28	vacmViewTreeFamilyType {vacmViewTreeFamilyEntry 4}	R/NC	[Standard] Indicates inclusion or exclusion of the MIB view: The default value is included. ● included (1) ● excluded (2) [Implementation] Same as the standard.	Y

#	Object identifier	Access	Implementation specifications	Implemented
29	vacmViewTreeFamilyStorageType {vacmViewTreeFamilyEntry 5}	R/NC	[Standard] The saving format of this entry. The default value is nonVolatile. [Implementation] Fixed value of readOnly (5).	Y
30	vacmViewTreeFamilyStatus {vacmViewTreeFamilyEntry 6}	R/NC	[Standard] Status of this entry. [Implementation] Same as the standard. Fixed value of active (1).	Y

3. Private MIBs

This chapter describes the implementation specifications for the private MIBs used by the Switch.

3.1 axsStats group (statistics MIB)

3.1.1 axsIfStats group

(1) ID

```
axsStats OBJECT IDENTIFIER ::= {axsMib 1}

axsIfStats OBJECT IDENTIFIER ::= {axsStats 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.1.4
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsIfStats group.

Table 3-1 axsIfStats group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsIfStatsTable {axsIfStats 1}	NOT-ACCESSIBLE	NA	Extended statistics table of the interface	Y
2	axsIfStatsEntry {axsIfStatsTable 1}	NOT-ACCESSIBLE	NA	Extended statistics table entry of the interface. INDEX {axsIfStatsIndex}	Y
3	axsIfStatsIndex {axsIfStatsEntry 1}	NOT-ACCESSIBLE	NA	Interface index of the Switch. Same as ifIndex.	Y
4	axsIfStatsName {axsIfStatsEntry 2}	DisplayString	R/O	Interface name. Same as ifDescr.	Y
5	axsIfStatsInMegaOctets {axsIfStatsEntry 3}	Counter	R/O	Total number (in millions) of octets received. Any fraction is truncated.	Y
6	axsIfStatsInUcastMegaPkts {axsIfStatsEntry 4}	Counter	R/O	Number (in millions) of unicast packets received. Any fraction is truncated.	Y
7	axsIfStatsInMulticastMegaPkts {axsIfStatsEntry 5}	Counter	R/O	Number (in millions) of multicast packets received. Any fraction is truncated.	Y
8	axsIfStatsInBroadcastMegaPkts {axsIfStatsEntry 6}	Counter	R/O	Number (in millions) of broadcast packets received. Any fraction is truncated.	Y
9	axsIfStatsOutMegaOctets {axsIfStatsEntry 7}	Counter	R/O	Total number (in millions) of octets sent. Any fraction is truncated.	Y
10	axsIfStatsOutUcastMegaPkts {axsIfStatsEntry 8}	Counter	R/O	Number (in millions) of packets sent by unicast. Any fraction is truncated.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
11	axsIfStatsOutMulticastMegaPkts {axsIfStatsEntry 9}	Counter	R/O	Number (in millions) of packets sent in multicast routing. Any fraction is truncated.	Y
12	axsIfStatsOutBroadcastMegaPkts {axsIfStatsEntry 10}	Counter	R/O	Number (in millions) of packets sent in broadcast. Any fraction is truncated.	Y
13	axsIfStatsHighSpeed {axsIfStatsEntry 11}	Counter	R/O	Speed of the line (in Mbit/s). Any fraction is truncated. If the bandwidth configuration command is not defined, the speed of the relevant line is displayed. If it is defined, the defined value is displayed.	Y

Note: Information pertains to the Ethernet interface.

3.1.2 axsQoS group

(1) ID

```
axsStats OBJECT IDENTIFIER ::= {axsMib 1}

axsQoS OBJECT IDENTIFIER ::= {axsStats 6}

axsEtherTxQoS OBJECT IDENTIFIER ::= {axsQoS 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.1.6.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsEtherTxQoS group.

Table 3-2 axsEtherTxQoS group implementation specifications (QoS statistics of Ethernet interface)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsEtherTxQoSStatsTable {axsEtherTxQoS 1}	NOT-ACCESSIBLE	NA	Table information on the QoS statistics.	Y
2	axsEtherTxQoSStatsEntry {axsEtherTxQoSStatsTable 1}	NOT-ACCESSIBLE	NA	Entry for the QoS statistics of each Ethernet interface. INDEX {axsEtherTxQoSStatsIndex}	Y
3	axsEtherTxQoSStatsIndex {axsEtherTxQoSStatsEntry 1}	NOT-ACCESSIBLE	NA	Shows the index value for identifying any entry in the table (ifIndex of the Ethernet interface). The value ranges from 1 to ifNumber.	Y
4	axsEtherTxQoSStatsMaxQnum {axsEtherTxQoSStatsEntry 2}	INTEGER	R/O	Returns the maximum number of queues of the relevant interface.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
5	axsEtherTxQoSStatsLimitQlen {axsEtherTxQoSStatsEntry 3}	INTEGER	R/O	Shows the limit length of the output priority queue of the relevant interface.	Y
6	axsEtherTxQoSStatsTotalOutFrames {axsEtherTxQoSStatsEntry 4}	Counter	R/O	Shows the total number of frames sent from the relevant interface. ● Fixed value of 0.	M
7	axsEtherTxQoSStatsTotalOutBytesHigh {axsEtherTxQoSStatsEntry 5}	Counter	R/O	Shows the total number of bytes sent from the relevant interface (most significant 4 bytes). ● Fixed value of 0.	M
8	axsEtherTxQoSStatsTotalOutBytesLow {axsEtherTxQoSStatsEntry 6}	Counter	R/O	Shows the total number of bytes sent from the relevant interface (least significant 4 bytes). ● Fixed value of 0.	M
9	axsEtherTxQoSStatsTotalDiscardFrames {axsEtherTxQoSStatsEntry 7}	Counter	R/O	Shows the total number of discarded frames in the relevant interface.	Y
10	axsEtherTxQoSStatsQueueTable {axsEtherTxQoS 2}	NOT-ACCESSIBLE	NA	Table information of the QoS statistics for each output priority queue of the relevant interface.	Y
11	axsEtherTxQoSStatsQueueEntry {axsEtherTxQoSStatsQueueTable 1}	NOT-ACCESSIBLE	NA	Entry of the QoS statistics for each output priority queue of the relevant interface. INDEX {axsEtherTxQoSStatsQueueIndex, axsEtherTxQoSStatsQueueQueIndex}	Y
12	axsEtherTxQoSStatsQueueIndex {axsEtherTxQoSStatsQueueEntry 1}	NOT-ACCESSIBLE	NA	Shows the index value for identifying any entry in the table (ifIndex of the Ethernet interface). The value ranges from 1 to ifNumber.	Y
13	axsEtherTxQoSStatsQueueQueIndex {axsEtherTxQoSStatsQueueEntry 2}	NOT-ACCESSIBLE	NA	Shows the index value for identifying any entry in the table. The value ranges from 1 to axsEtherTxQoSStatsMaxQnum.	Y
14	axsEtherTxQoSStatsQueueQlen {axsEtherTxQoSStatsQueueEntry 3}	INTEGER	R/O	Shows the length of the output priority queue at the time of information collection. ● Fixed value of 0.	M

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
15	axsEtherTxQoSStatsQueueMaxQlen {axsEtherTxQoSStatsQueueEntry 4}	INTEGER	R/O	Shows the maximum length of the output priority queue of the relevant interface after the statistics are deleted or initialized. ● Fixed value of 0.	M
16	axsEtherTxQoSStatsQueueDiscardFramesClass1 {axsEtherTxQoSStatsQueueEntry 5}	Counter64	R/O	Shows the number of discarded frames in the relevant output priority queue at queuing level 1. ● Fixed value of 0.	M
17	axsEtherTxQoSStatsQueueDiscardFramesClass2 {axsEtherTxQoSStatsQueueEntry 6}	Counter64	R/O	Shows the number of discarded frames in the relevant output priority queue at queuing level 2. ● Fixed value of 0.	M
18	axsEtherTxQoSStatsQueueDiscardFramesClass3 {axsEtherTxQoSStatsQueueEntry 7}	Counter64	R/O	Shows the number of discarded frames in the relevant output priority queue at queuing level 3. ● Fixed value of 0.	M
19	axsEtherTxQoSStatsQueueDiscardFramesClass4 {axsEtherTxQoSStatsQueueEntry 8}	Counter64	R/O	Shows the number of discarded frames in the relevant output priority queue at queuing level 4. ● Fixed value of 0.	M

3.2 axsFdb group (MAC address table group MIB)

(1) ID

```
axsMib OBJECT IDENTIFIER ::= {axsEx 1}

axsFdb OBJECT IDENTIFIER ::= {axsMib 5}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.5
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsFdb group.

Table 3-3 axsFdb group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsFdbCounterTable {axsFdb 1}	NOT-ACCESSIBLE	NA	Table of the MAC address learning count.	Y
2	axsFdbCounterEntry {axsFdbCounterTable 1}	NOT-ACCESSIBLE	NA	Entry of the table of the MAC address learning count. INDEX {axsFdbCounterNifIndex,axsFdbCounterLineIndex}	Y
3	axsFdbCounterNifIndex {axsFdbCounterEntry 1}	NOT-ACCESSIBLE	NA	Shows the position of slots with NIF inserted.	Y
4	axsFdbCounterLineIndex {axsFdbCounterEntry 2}	NOT-ACCESSIBLE	NA	Shows the position of slots with LINE inserted.	Y
5	axsFdbCounterCounts {axsFdbCounterEntry 3}	Counter32	R/O	Number of table entries of the MAC address learning count at this port.	Y
6	axsFdbCounterType {axsFdbCounterEntry 4}	INTEGER	R/O	Setting of the learning limitation, and behavior of unlearnt frames in case of MAC address table count overflow when limitation is valid: ● Unlimited (0) ● Limited and Forward (1) ● Limited and Discard (2)	Y
7	axsFdbCounterLimits {axsFdbCounterEntry 5}	Counter32	R/O	Maximum number of table entries for the MAC address learning at this port.# 0: Learning prohibited. From 1 to 100000: Number specified via the `mac-address-table static` configuration command.	Y

Note: Fixed value of 0 when axsFdbCounterType is Unlimited (0).

3.3 axsVlan group (VLAN information MIB)

The port number used in this group is a number that uniquely distinguishes the physical port number, the channel group number, or the virtual link ID, and it is calculated by using the following relationships:

- Port number of a physical port

  Any physical ports are uniquely identified.

  Port number: Physical port number

- Port number of a channel group

  Calculated from the channel group number in link aggregation as follows:

  Port number: 65 (fixed value) + channel group number

- Port number of a virtual link

  Calculated from the virtual link ID as follows:

  Port number: 193 (fixed value) + virtual link ID

  A port number of a virtual link is applied only to the axsVBStpPortTable group.

3.3.1 axsVlanBridge group (dot1dBase information)

(1) axsVBBaseTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeBase OBJECT IDENTIFIER ::= {axsVlanBridge 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.1
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBBaseTable group.

Table 3-4 axsVBBaseTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBBaseTable {axsVlanBridgeBase 1}	NOT-ACCESSIBLE	NA	Table of dot1dBase information for each VLAN.	Y
2	axsVBBaseEntry {axsVBBaseTable 1}	NOT-ACCESSIBLE	NA	Entry of information on each VLAN ID in axsVBBaseTable. INDEX {axsVBBaseIndex}	Y
3	axsVBBaseIndex {axsVBBaseEntry 1}	VlanIndex	R/O	VLAN ID	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	axsVBBaseBridgeAddress {axsVBBaseEntry 2}	MacAddress	R/O	MAC address of the VLAN. When using MAC functionality for each VLAN: MAC address for each VLAN. When not using the above: MAC address of the Switch.	Y
5	axsVBBaseNumPorts {axsVBBaseEntry 3}	INTEGER	R/O	Number of ports assigned in the VLAN	Y
6	axsVBBaseType {axsVBBaseEntry 4}	INTEGER	R/O	Type of bridging the VLAN can execute: ● unknown (1) ● transparent-only (2) ● sourceroute-only (3) ● srt (4) Returns a fixed value of transparent-only (2) in the Switch.	M
7	axsVBBaseVlanIfIndex {axsVBBaseEntry 5}	INTEGER	R/O	Value of ifIndex of the VLAN interface.	Y
8	axsVBBaseVlanType {axsVBBaseEntry 6}	INTEGER	R/O	VLAN type. ● port-based (1) ● mac-based (2) ● protocol-based (3)	Y
9	axsVBBaseVlanID {axsVBBaseEntry 7}	VlanIdOrZero	R/O	The VLAN ID value of the VLAN tag corresponding to the VLAN.	Y
10	axsVBBaseAssociatedPrimaryVlan {axsVBBaseEntry 8}	VlanIdOrZero	R/O	Returns the VLAN ID of the primary VLAN paired with this VLAN when private VLAN functionality is active and this VLAN is set as a secondary VLAN. Returns 0 when private VLAN functionality is not active in this VLAN, this VLAN is not a secondary VLAN, or there is no primary VLAN paired with this VLAN. Returns a fixed value (0) in the Switch.	Y
11	axsVBBaseIfStatus {axsVBBaseEntry 9}	INTEGER	R/O	Interface state of the VLAN against the upper protocol: ● Up (1) ● Down (2)	Y
12	axsVBBaseLastChange {axsVBBaseEntry 10}	TimeTicks	R/O	Value of sysUpTime when the VLAN topology is changed.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
13	axsVBBasePrivateVlanType {axsVBBaseEntry 11}	INTEGER	R/O	Private VLAN type of VLAN. Returns normal (1) when the private VLAN functionality is not used. ● normal (1) ● primary (2) ● isolated (3) ● community (4) Returns a fixed value (1) in the Switch.	Y

(2) axsVBBasePortTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeBase OBJECT IDENTIFIER ::= {axsVlanBridge 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.1
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBBasePortTable group.

Table 3-5 axsVBBasePortTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBBasePortTable {axsVlanBridgeBase 2}	NOT-ACCESSIBLE	NA	Table of dot1dBasePortTable for each VLAN.	Y
2	axsVBBasePortEntry {axsVBBasePortTable 1}	NOT-ACCESSIBLE	NA	Structural entry of axsVBBasePortTable. INDEX {axsVBBasePortIndex, axsVBBasePort}	Y
3	axsVBBasePortIndex {axsVBBasePortEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBBasePort {axsVBBasePortEntry 2}	INTEGER	R/O	Port number (1-65535) of the port assigned in the VLAN. Such port numbers are assigned to physical ports and link aggregation.	Y
5	axsVBBasePortIfIndex {axsVBBasePortEntry 3}	INTEGER	R/O	Value of ifIndex for the port assigned in the VLAN.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	axsVBBasePortCircuit {axsVBBasePortEntry 4}	OBJECT IDENTIFIER	R/O	ID to distinguish different ports in the VLAN that have the same axsVBBasePortIfIndex. Returns a fixed value (0.0) in the Switch.	M
7	axsVBBasePortDelayExceededDiscards {axsVBBasePortEntry 5}	Counter	R/O	Number of discarded frames due to a delay occurring in a VLAN port. Returns a fixed value (0) in the Switch.	M
8	axsVBBasePortMtuExceededDiscards {axsVBBasePortEntry 6}	Counter	R/O	Number of discarded frames due to data overflow occurring in a VLAN port. Returns a fixed value (0) in the Switch.	M
9	axsVBBasePortState {axsVBBasePortEntry 7}	INTEGER	R/O	STP port state of a port assigned in the VLAN: ● disable (1) ● blocking (2) ● listening (3) ● learning (4) ● forwarding (5) ● broken (6) ● fix-forwarding (7) Returns disable (1), blocking (2), listening (3), learning (4), forwarding (5), or fix-forwarding (7) in the Switch.	Y
10	axsVBBasePortTaggedState {axsVBBasePortEntry 8}	INTEGER	R/O	State of the VLAN tag configuration of the port set as a VLAN ● Without configuration (1) ● With configuration (2)	Y
11	axsVBBasePortTranslatedTagID {axsVBBasePortEntry 9}	VlanIdOrZero	R/O	Translation ID (1-4095) of the VLAN assigned to this port when tag translation is active. Returns (0) when tag translation is not active.	Y

(3) axsVBStpTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1
```

```
axsVlanBridgeStp OBJECT IDENTIFIER ::= {axsVlanBridge 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.2
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBStpTable group.

Table 3-6 axsVBStpTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBStpTable {axsVlanBridgeStp 1}	NOT-ACCESSIBLE	NA	Table of dot1dStpTable for each VLAN. This table contains Spanning Tree information for each PVST+ VLAN.	Y
2	axsVBStpEntry {axsVBStpTable 1}	NOT-ACCESSIBLE	NA	Entry of axsVBStpTable. INDEX {axsVBStpIndex}	Y
3	axsVBStpIndex {axsVBStpEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBStpProtocolSpecification {axsVBStpEntry 2}	INTEGER	R/O	Type of Spanning Tree Protocol for each VLAN: ● unknown (1) ● decLb100 (2) ● ieee8021d (3) ● ieee8021w (4) Returns ieee8021d (3) or ieee8021w (4) in the Switch.	Y
5	axsVBStpPriority {axsVBStpEntry 3}	INTEGER	R/O	Priority value of the Spanning Tree Protocol for each VLAN (0-65535).	Y
6	axsVBStpTimeSinceTopologyChange {axsVBStpEntry 4}	TimeTicks	R/O	Elapsed time since the change in the topology of the Spanning Tree Protocol for each VLAN (in units of 1/100 seconds).	Y
7	axsVBStpTopChanges {axsVBStpEntry 5}	Counter	R/O	Number of topology changes in the Spanning Tree Protocol for each VLAN.	Y
8	axsVBStpDesignatedRoot {axsVBStpEntry 6}	BridgeId	R/O	Root bridge ID of a Spanning Tree Protocol for each VLAN.	Y
9	axsVBStpRootCost {axsVBStpEntry 7}	INTEGER	R/O	Root path cost of a Spanning Tree Protocol for each VLAN.	Y
10	axsVBStpRootPort {axsVBStpEntry 8}	INTEGER	R/O	Root port value of a Spanning Tree Protocol for each VLAN.	Y
11	axsVBStpMaxAge {axsVBStpEntry 9}	Timeout	R/O	Maximum aging time of a Spanning Tree Protocol for each VLAN (in units of 1/100 seconds).	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
12	axsVBStpHelloTime {axsVBStpEntry 10}	Timeout	R/O	Hello time of a Spanning Tree Protocol for each VLAN (in units of 1/100 seconds).	Y
13	axsVBStpHoldTime {axsVBStpEntry 11}	INTEGER	R/O	Hold time of a Spanning Tree for each VLAN (in units of 1/100 seconds).	Y
14	axsVBStpForwardDelay {axsVBStpEntry 12}	Timeout	R/O	Transfer delay time of a Spanning Tree Protocol for each VLAN (in units of 1/100 seconds).	Y
15	axsVBStpBridgeMaxAge {axsVBStpEntry 13}	Timeout	R/O	Maximum aging time of the Spanning Tree Protocol (for each VLAN) that is set up as a root bridge (range 600 to 4000, unit: 1/100 seconds).	Y
16	axsVBStpBridgeHelloTime {axsVBStpEntry 14}	Timeout	R/O	Hello time of the Spanning Tree Protocol for each VLAN that is set up as a root bridge (range 100 to 1000, unit: 1/100 seconds).	Y
17	axsVBStpBridgeForwardDelay {axsVBStpEntry 15}	Timeout	R/O	Transfer delay time of the Spanning Tree Protocol for each VLAN that is set up as a root bridge (range 400 to 3000, unit: 1/100 seconds).	Y

(4) axsVBStpPortTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeStp OBJECT IDENTIFIER ::= {axsVlanBridge 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.2
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBStpPortTable group.

Table 3-7 axsVBStpPortTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBStpPortTable {axsVlanBridgeStp 2}	NOT-ACCESSIBLE	NA	Table of dot1dStpPortTable for each VLAN. This table contains the information on Spanning Tree ports for each PVST+ VLAN.	Y
2	axsVBStpPortEntry {axsVBStpPortTable 1}	NOT-ACCESSIBLE	NA	Structural entry of axsVBStpPortTable. INDEX {axsVBStpPortIndex, axsVBStpPort}	Y
3	axsVBStpPortIndex {axsVBStpPortEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBStpPort {axsVBStpPortEntry 2}	INTEGER	R/O	Port number (1-65535) of the port corresponding to the structural entry. Such port numbers are assigned to physical ports, link aggregation, and virtual links.	Y
5	axsVBStpPortPriority {axsVBStpPortEntry 3}	INTEGER	R/O	Priority of this port for each VLAN (0-255).	Y
6	axsVBStpPortState {axsVBStpPortEntry 4}	INTEGER	R/O	Current state of this port for each VLAN: ● disabled (1) ● blocking (2) ● listening (3) ● learning (4) ● forwarding (5) ● broken (6) Returns disabled (1), blocking (2), listening (3), learning (4), or forwarding (5) in the Switch.	Y
7	axsVBStpPortEnable {axsVBStpPortEntry 5}	INTEGER	R/O	Shows whether the Spanning Tree functionality for each VLAN is enabled or disabled for this port: ● enabled (1) ● disabled (2)	Y
8	axsVBStpPortPathCost {axsVBStpPortEntry 6}	INTEGER	R/O	Path cost of this port for each VLAN (1-200000000). Displays 0 if the port link is down.	Y
9	axsVBStpPortDesignatedRoot {axsVBStpPortEntry 7}	BridgeId	R/O	Root bridge ID value for each VLAN included in the BPDU received by this port from the designated bridge.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
10	axsVBStpPortDesignatedCost {axsVBStpPortEntry 8}	INTEGER	R/O	Path cost for each VLAN of the designated port connected to this port.	Y
11	axsVBStpPortDesignatedBridge {axsVBStpPortEntry 9}	BridgeId	R/O	Bridge ID of the port for each VLAN regarded as a designated bridge for this port.	Y
12	axsVBStpPortDesignatedPort {axsVBStpPortEntry 10 }	OCTET STRING	R/O	Port ID of the designated bridge for each VLAN connected to this port.	Y
13	axsVBStpPortForwardTransitions {axsVBStpPortEntry 11}	Counter	R/O	Number of times this port has changed its state from learning to forwarding for each VLAN.	Y

(5) axsVBTpTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeTp OBJECT IDENTIFIER ::= {axsVlanBridge 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.4
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBTpTable group.

Table 3-8 axsVBTpTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBTpTable {axsVlanBridgeTp 1}	NOT-ACCESSIBLE	NA	dot1dTp information table for each VLAN.	Y
2	axsVBTpEntry {axsVBTpTable 1}	NOT-ACCESSIBLE	NA	Structural entry of axsVBTpTable. INDEX {axsVBTpIndex}	Y
3	axsVBTpIndex {axsVBTpEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBTpLearnedEntryDiscards {axsVBTpEntry 2}	Counter	R/O	Number of entries discarded due to the absence of free space in the MAC address table. Returns a fixed value (0) in the Switch.	M

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
5	axsVBTpAgingTime {axsVBTpEntry 3}	INTEGER	R/O	Timeout period to age out dynamically learned MAC address entries (in seconds). ● In aging mode: 10 to 1000000 ● Not in aging mode: 0	Y

(6) axsVBTpFdbTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeTp OBJECT IDENTIFIER ::= {axsVlanBridge 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.4
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBTpFdbTable group.

Table 3-9 axsVBTpFdbTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBTpFdbTable {axsVlanBridgeTp 2}	NOT-ACCESSIBLE	NA	dot1dTpFdbTable information table for each VLAN.	Y
2	axsVBTpFdbEntry {axsVBTpFdpTable 1}	NOT-ACCESSIBLE	NA	Structural entry of axsVBTpFdbTable. INDEX {axsVBTpFdbIndex,axsVBTpFdbAddress}	Y
3	axsVBTpFdbIndex {axsVBTpEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBTpFdbAddress {axsVBTpEntry 2}	MacAddress	R/O	Unicast MAC address in the entries in the MAC address table.	Y
5	axsVBTpFdbPort {axsVBTpEntry 3}	INTEGER	R/O	Port number that received frames whose source address is the MAC address designated by axsVBTpFdbAddress. If 0, no port number is learned.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	axsVBTpFdbStatus {axsVBTpEntry 4}	INTEGER	R/O	State of the MAC address table: ● other (1) ● invalid (2) ● learned (3) ● self (4) ● mgmt (5) Returns learned (3) for dynamic entries. Returns mgmt (5) for static entries.	Y

(7) axsVBTpPortTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeTp OBJECT IDENTIFIER ::= {axsVlanBridge 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.4
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBTpPortTable group.

Table 3-10 axsVBTpPortTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBTpPortTable {axsVlanBridgeTp 3}	NOT-ACCESSIBLE	NA	dot1dTpPortTable information table for each VLAN.	Y
2	axsVBTpPortEntry {axsVBTpPortTable 1}	NOT-ACCESSIBLE	NA	Entry of axsVBTpPortTable information for each port. INDEX {axsVBTpPortIndex,axsVBTpPort}	Y
3	axsVBTpPortIndex {axsVBTpPortEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBTpPort {axsVBTpPortEntry 2}	INTEGER	R/O	Port number (1-65535) that indicates the port corresponding to the management information in this entry. Such port numbers are assigned to physical ports and link aggregation.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
5	axsVBTpPortMaxInfo {axsVBTpPortEntry 3}	INTEGER	R/O	Maximum width of the INFO field of this port for each VLAN (excluding the MAC header and FCS).	Y
6	axsVBTpPortInFrames {axsVBTpPortEntry 4}	Counter	R/O	Number of frames received by this port for each VLAN. Returns a fixed value (0) in the Switch.	M
7	axsVBTpPortOutFrames {axsVBTpPortEntry 5}	Counter	R/O	Number of frames sent by this port for each VLAN. Returns a fixed value (0) in the Switch.	M
8	axsVBTpPortInDiscards {axsVBTpPortEntry 6}	Counter	R/O	Number of frames discarded in this port for each VLAN. Returns a fixed value (0) in the Switch.	M

(8) axsVBStaticTable group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1

axsVlanBridgeStatic OBJECT IDENTIFIER ::= {axsVlanBridge 5}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1.5
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVBStaticTable group.

Table 3-11 axsVBStaticTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVBStaticTable {axsVlanBridgeStatic 1}	NOT-ACCESSIBLE	NA	dot1dStaticTable information table for each VLAN.	Y
2	axsVBStaticEntry {axsVBStaticTable 1}	NOT-ACCESSIBLE	NA	Structural entry of axsVBStaticTable. INDEX {axsVBStaticIndex,axsVBStaticAddress }	Y
3	axsVBStaticIndex {axsVBStaticEntry 1}	VlanIndex	R/O	VLAN ID	Y
4	axsVBStaticAddress {axsVBStaticEntry 2}	MacAddress	R/O	MAC address for unicast, group, or broadcast.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
5	axsVBStaticReceivePort {axsVBStaticEntry 3}	INTEGER	R/O	The receiving port number that this entry is applied to. 0 if applied to all receiving ports. Returns a fixed value (0) in the Switch.	Y
6	axsVBStaticAllowedToGoTo {axsVBStaticEntry 4}	OCTET STRING	R/O	Bitmap of a port that indicates the destination port of the forwarding of the frame whose destination is the MAC address included in this entry.	Y
7	axsVBStaticStatus {axsVBStaticEntry 5}	INTEGER	R/O	State of this entry: ● other (1) ● invalid (2) ● permanent (3) ● deleteOnReset (4) ● deleteOnTimeout (5) In the Switch, returns permanent (3) for static entries and deleteOnReset (4) for IGMP/MLD snooping entries.	Y

(9) axsVlanBridge (others) group

(a) ID

```
axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanBridge OBJECT IDENTIFIER ::= {axsVlan 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.1
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVlanBridge (others) group.

Table 3-12 axsVlanBridge (others) group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVlanBridgeMaxVlans {axsVlanBridge 101}	VlanIndex	R/O	Maximum value of the VLAN IDs in the Switch Returns a fixed value (4095) in the Switch.	Y
2	axsVlanBridgeMaxSpans {axsVlanBridge 102}	VlanIndex	R/O	Maximum value of the VLAN IDs of the VLANs where Spanning Tree Protocols are operated in the Switch Returns a fixed value (4095) in the Switch.	Y

3.3.2 axsVlanTagTranslation group (Tag translation information MIB)

(a) ID

```
 axsVlan OBJECT IDENTIFIER ::= {axsMib 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6

axsVlanTagTranslation OBJECT IDENTIFIER ::= {axsVlan 10}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.10

axsVlanTagTranslationTable OBJECT IDENTIFIER ::= {axsVlanTagTranslation 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.6.10.1
```

(b) Implementation specifications

The following table shows the implementation specifications for the axsVlanTagTranslation group.

Table 3-13 axsVlanTagTranslation group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsVlanTagTranslation Table {axsVlanTagTranslation 1}	NOT-ACCESSIBLE	NA	Information table about tag translation.	Y
2	axsVlanTagTranslation Entry {axsVlanTagTranslation Table 1}	NOT-ACCESSIBLE	NA	Entries in the information about tag translation. INDEX { axsVlanTagTranslationVlanId, axsVlanTagTranslationTranslatedId }	Y
3	axsVlanTagTranslation VlanId {axsVlanTagTranslation Entry 1}	NOT-ACCESSIBLE	NA	VLAN ID (1-4094) to which tag translation is assigned.	Y
4	axsVlanTagTranslation TranslatedId {axsVlanTagTranslation Entry 2}	NOT-ACCESSIBLE	NA	Translated ID (1-4094) configured in the tag transation.	Y
5	axsVlanTagTranslation Ports {axsVlanTagTranslation Entry 3}	PortList	R/O	List of ports that assign the same translated IDs to the same VLANs via tag translation.	Y

3.4 axsL2ldMIB group (L2LD information MIB)

3.4.1 axsL2ldGlobalInfo group

(1) ID

```
axsL2ld OBJECT IDENTIFIER ::= {axsMib 10}

axsL2ldGlobalInfo OBJECT IDENTIFIER ::= {axsL2ld 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.10.1

axsL2ldVersion OBJECT IDENTIFIER ::= {axsL2ldGlobalInfo 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.10.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsL2ldGlobalInfo group.

Table 3-14 axsL2ldGlobalInfo group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsL2ldVersion {axsL2ldGlobalInfo 1}	INTEGER	R/O	Version of L2 loop detection. ● Version 1 (1)	Y
2	axsL2ldLoopDetectionId {axsL2ldGlobalInfo 2}	INTEGER	R/O	ID of L2 loop detection. ● Fixed value of 0.	Y
3	axsL2ldIntervalTime {axsL2ldGlobalInfo 3}	INTEGER	R/O	Sending interval of L2 loop detection frames (in seconds).	Y
4	axsL2ldOutputRate {axsL2ldGlobalInfo 4}	INTEGER	R/O	Transmission rate of L2 loop detection frames (in packets per second).	Y
5	axsL2ldThreshold {axsL2ldGlobalInfo 5}	INTEGER	R/O	Number of detections until the port changes to the inactive state.	Y
6	axsL2ldHoldTime {axsL2ldGlobalInfo 6}	INTEGER	R/O	Retention time for the number of detections (in seconds).	Y
7	axsL2ldAutoRestoreTime {axsL2ldGlobalInfo 7}	INTEGER	R/O	Period to switch an inactive port to an active port automatically (in seconds).	Y
8	axsL2ldConfigurationVlanPortCounts {axsL2ldGlobalInfo 8}	INTEGER	R/O	Number of VLAN ports that are directed to send L2 loop detection frames.	Y
9	axsL2ldCapacityVlanPortCounts {axsL2ldGlobalInfo 9}	INTEGER	R/O	Number of VLAN ports that are able to send L2 loop detection frames at the defined transmission rate.	Y

3.4.2 axsL2ldPortTable group

(1) ID

```
axsL2ld OBJECT IDENTIFIER ::= {axsMib 10}

axsL2ldPortTable Group OBJECT IDENTIFIER ::= {axsL2ld 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.10.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsL2ldPortTable group.

Table 3-15 axsL2ldPortTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsL2ldPortTable {axsL2ld 2}	NOT-ACCESSIBLE	NA	Table containing the port information about L2 loop detection.	Y
2	axsL2ldPortEntry {axsL2ldPortTable 1}	NOT-ACCESSIBLE	NA	List of the port information about L2 loop detection. INDEX {axsL2ldPortIndex,axsL2ldPortIfIndex }	Y
3	axsL2ldPortIndex {axsL2ldPortEntry 1}	INTEGER	R/O	Fixed value of 1.	Y
4	axsL2ldPortIfIndex {axsL2ldPortEntry 2}	INTEGER	R/O	ifIndex[#1#2] of the port.	Y
5	axsL2ldPortStatus {axsL2ldPortEntry 3}	INTEGER	R/O	Port status: For ports, and link aggregation: ● Up (1): Up state ● Down (2): Down state ● Down (loop) (3): Down state due to L2 loop detection For peer link: ● Up (1): Full state ● Down (2): Standalone or Conflict state	Y
6	axsL2ldPortType {axsL2ldPortEntry 4}	INTEGER	R/O	Port type: ● trap (1): Detecting port ● send-inact (2): Detecting and blocking port ● send (3): Detecting and sending port ● uplink (4): Uplink port pair ● exception (5): Ports exempted from detection	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
7	axsL2ldPortDetectCount {axsL2ldPortEntry 5}	INTEGER	R/O	Number of L2 loop detections. How many times the L2 loop detection frames are received within the retention time.	Y
8	axsL2ldPortAutoRestoringTimer {axsL2ldPortEntry 6}	INTEGER	R/O	Time to automatic recovery (in seconds). 0 if the port is active.	Y
9	axsL2ldPortSourcePortIfindex {axsL2ldPortEntry 7}	INTEGER	R/O	ifIndex#1#2 of the port that sent the last L2 loop detection frame received.	Y
10	axsL2ldPortDestinationPortIfindex {axsL2ldPortEntry 8}	INTEGER	R/O	ifIndex#1#2 of the port that received the last L2 loop detection frame.	Y
11	axsL2ldPortSourceVlan {axsL2ldPortEntry 9}	INTEGER	R/O	VLAN ID of the sender of the last L2 loop detection frame received	Y
12	axsL2ldPortHCInFrames {axsL2ldPortEntry 10}	Counter64	R/O	Number of L2 loop detection frames received.	Y
13	axsL2ldPortHCOutFrames {axsL2ldPortEntry 11}	Counter64	R/O	Number of L2 loop detection frames sent.	Y
14	axsL2ldPortHCInDiscards {axsL2ldPortEntry 12}	Counter64	R/O	Number of received L2 loop detection frames discarded	Y
15	axsL2ldPortInactiveCount {axsL2ldPortEntry 13}	INTEGER	R/O	Number of times the port changed to the inactive state.	Y
16	axsL2ldPortLastInactiveTime {axsL2ldPortEntry 14}	TimeStamp	R/O	Start time of the last inactive state.	Y
17	axsL2ldPortLastInFramesTime {axsL2ldPortEntry 15}	TimeStamp	R/O	Time when the last L2 loop detection frame is received.	Y

#1: For a link aggregation port, ifIndex from link aggregation is used.

#2: For a peer link port, ifIndex from the peer link is used.

3.5 axsUlr group (uplink redundancy information MIB)

3.5.1 axsUlrGlobalInfo group

(1) ID

```
axsUlr OBJECT IDENTIFIER ::= {axsMib 20}

axsUlrGlobalInfo OBJECT IDENTIFIER ::= {axsUlr 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.20.1

axsUlrVersion OBJECT IDENTIFIER ::= {axsUlrGlobalInfo 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.20.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsUlrGlobalInfo group.

Table 3-16 axsUlrGlobalInfo group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsUlrVersion {axsUlrGlobalInfo 1}	INTEGER	R/O	Uplink redundancy version. ● Version 1(1)	Y
2	axsUlrID {axsUlrGlobalInfo 2}	MacAddress	R/O	System ID of the switch.	Y
3	axsUlrConfigurationPortCounts {axsUlrGlobalInfo 3}	INTEGER	R/O	Summed number of primary ports and secondary ports.	Y
4	axsUlrStartupActivePortSelection {axsUlrGlobalInfo 4}	INTEGER	R/O	Setting of the functionality to fix the active port at Switch startup. ● Off (1): No ● On (2): Yes	N

3.5.2 axsUlrPortTable group

(1) ID

```
axsUlr OBJECT IDENTIFIER ::= {axsMib 20}

axsUlrPortTable Group OBJECT IDENTIFIER ::= {axsUlr 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.20.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsUlrPortTable group:

Table 3-17 axsUlrPortTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsUlrPortTable {axsUlr 2}	NOT-ACCESSIBLE	NA	Table for storing port information for uplink redundancy.	Y
2	axsUlrPortEntry {axsUlrPortTable 1}	NOT-ACCESSIBLE	NA	List of port information for uplink redundancy. INDEX { axsUlrPortIfIndex }	Y
3	axsUlrPortIfIndex {axsUlrPortEntry 1}	INTEGER	R/O	ifIndex of ports or channel groups.	Y
4	axsUlrPortType {axsUlrPortEntry 2}	INTEGER	R/O	Port type: ● Primary (1): Primary port ● Secondary (2): Secondary port	Y
5	axsUlrPairedPortIfIndex {axsUlrPortEntry 3}	INTEGER	R/O	ifIndex of paired ports or paired channel groups.	Y
6	axsUlrPortStatus {axsUlrPortEntry 4}	INTEGER	R/O	Port status: ● Forwarding (1): Forwarding ● Down (2): Port or channel group is down. ● Blocking (3): Blocking	Y
7	axsUlrPairedPortStatus {axsUlrPortEntry 5}	INTEGER	R/O	Status of paired ports. ● Forwarding (1): Forwarding ● Down (2): Ports or channel groups are down. ● Blocking (3): Blocking	N
8	axsUlrAutoChangeToPrimary {axsUlrPortEntry 6}	INTEGER	R/O	Setting of automatic switchback: ● Off (1): No ● On (2): Yes	N
9	axsUlrAutoChangeToPrimaryDelay {axsUlrPortEntry 7}	INTEGER	R/O	Time of automatic switchback (in seconds).	N
10	axsUlrAutoChangeToPrimaryRest {axsUlrPortEntry 8}	INTEGER	R/O	Remaining time until automatic switchback (in seconds).	N
11	axsUlrStartupActivePortSelectionStatus {axsUlrPortEntry 9}	INTEGER	R/O	Operating state of the functionality to fix the active port at Switch startup: ● Off (1): Turned off ● On (2): Operating	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
12	axsUlrFlushTransmit {axsUlrPortEntry 10}	INTEGER	R/O	Setting of sending flush control frames: ● Off (1): No ● On (2): Yes	N
13	axsUlrFlushVlan {axsUlrPortEntry 11}	INTEGER	R/O	VLAN ID of the VLAN that sends flush control frames: ● Not defined (0) ● VLAN ID used as sender	N
14	axsUlrMacAddressUpdateTransmit {axsUlrPortEntry 12}	INTEGER	R/O	Number of MAC address update frames sent	N
15	axsUlrLastActivePortDecisionTime {axsUlrPortEntry 13}	TimeStamp	R/O	Last decision time for defining the active port	N
16	axsUlrLastFlushTransmitTime {axsUlrPortEntry 14}	TimeStamp	R/O	Last transmission time of flush control frames	N
17	axsUlrLastMacUpdateTransmitTime {axsUlrPortEntry 15}	TimeStamp	R/O	Last transmission time of MAC address update frames	N
18	axsUlrLastChangeFactor {axsUlrPortEntry 16}	INTEGER	R/O	Factor for defining the last active port: ● command (1) ● configure (2) ● primary down (3) ● primary up (4) ● secondary down (5) ● secondary up (6) ● preemption (7)	N
19	axsUlrFlushTransmitTotalPackets {axsUlrPortEntry 17}	INTEGER	R/O	Number of flush control frames sent	N
20	axsUlrMacAddressUpdateTransmitTotalPackets {axsUlrPortEntry 18}	INTEGER	R/O	Number of MAC address update frames sent	N
21	axsUlrMacAddressUpdateTransmitOverFlow {axsUlrPortEntry 19}	INTEGER	R/O	Number of overflows of MAC address update frames	N
22	axsUlrActiveDecisionCount {axsUlrPortEntry 20}	INTEGER	R/O	Number of times an active port is defined on an uplink port (including the target port)	N

3.6 axsBootManagement group (system boot information MIB)

(1) ID

```
axsBootManagement OBJECT IDENTIFIER ::= {axsMib 51}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.51
```

(2) Implementation specifications

The following table shows the implementation specifications for axsBootManagement.

Table 3-18 axsBootManagement implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsBootReason {axsBootManagement 1}	INTEGER	R/O	Reason for system startup: ● power-on (1): Startup due to power-on ● reload (2): Startup due to a command ● system-fault (3): Startup due to a fault ● system-stall (4): Startup due to a WDT timeout ● reset (5): Startup due to a hardware reset ● fail-over (6): Startup due to a SWAP ● default-restart (7): Startup due to a default restart ● system-exception (8): Startup due to fault (CPU exception) ● wake-on-rtc (9): Startup of the device waking from a sleep state to normal ● wake-on-reset (10): Startup of the device in a sleep state when the **RESET** button is held down The device returns power-on (1), reload (2), system-fault (3), system-stall (4), or system-exception (8).	Y

3.7 axsLogin group (login information MIB)

(1) ID

```
axsLogin OBJECTIDENTIFIER ::= {axsMib 52}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.52
```

(2) Implementation specifications

The following table shows the implementation specifications for axsLogin.

Table 3-19 axsLogin implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsLoginName {axsLogin 1}	DisplayString	NA	Login user name	Y
2	axsLoginTime {axsLogin 2}	DisplayString	NA	Indicates the user login time (year, month, day, hour, minute, second, and time zone) with a 26-byte string. The time is displayed in the *YYYY*/*MM*/*DD hh*:*mm*:*ss XXXXXX* format. ● *YYYY*: domonical year ● *MM*: month (01-12) ● *DD*: day (01-31) ● *hh*: hour (00-23) ● *mm*: minute (00-59) ● *ss*: second (00-59) ● *XXXXXX*: time zone A 1-byte space is inserted between *DD* and *hh*, and between *ss* and *XXXXXX*. (Example: 2010/08/12 10:23:10 JST)	Y
3	axsLogoutTime {axsLogin 3}	DisplayString	NA	Indicates the user logout time (year, month, day, hour, minute, second, and time zone) with a 26-byte string. The time is displayed in the *YYYY*/*MM*/*DD hh*:*mm*:*ss XXXXXX* format. ● *YYYY*: domonical year ● *MM*: month (01-12) ● *DD*: day (01-31) ● *hh*: hour (00-23) ● *mm*: minute (00-59) ● *ss*: second (00-59) ● *XXXXXX*: time zone A 1-byte space is inserted between *DD* and *hh*, and between *ss* and *XXXXXX*. (Example: 2010/08/12 10:23:10 JST)	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	axsLoginFailureTime {axsLogin 4}	DisplayString	NA	Indicates the time of a failed log in (year, month, day, hour, minute, second, and time zone) with a 26-byte string. The time is shown in the *YYYY*/*MM*/*DD hh*:*mm*:*ss XXXXXX* format. ● *YYYY*: domonical year ● *MM*: month (01-12) ● *DD*: day (01-31) ● *hh*: hour (00-23) ● *mm*: minute (00-59) ● *ss*: second (00-59) ● *XXXXXX*: time zone A 1-byte space is inserted between *DD* and *hh*, and between *ss* and *XXXXXX*. (Example: 2010/08/12 10:23:10 JST)	Y
5	axsLoginLocation {axsLogin 5}	DisplayString	NA	Connection type of the login user, indicated by either of the following formats: ● For console: Displayed as "console" ● For any other: Displayed as "*XXXX*(*YYYY*)" ■ *XXXX*: IP addres ■ *YYYY*: Application (telnet, ftp) (Example: "console", "192.168.1.1(telnet)")	Y
6	axsLoginLine {axsLogin 6}	DisplayString	NA	Terminal ID of the login user, indicated by either of the following connection types: ● For console: Displayed as "console" ● For any other: Displayed as "*XXXX*(*YYYY*)" ■ *XXXX*: Terminal ■ *YYYY*: Application (telnet, ftp) (Example: "console", "vty0(telnet)")	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
7	axsLogoutStatus {axsLogin 7}	INTEGER	NA	Reason for logout: ● error (1): Any reason other than the following (Example: System out due to an internal error) ● success (2): Logged out by using a command ● timeout (3): Auto-logout ● disconnect (4): Disconnection ● force (5): Forcibly logged out by another user Returns a fixed value (2) in the Switch.	Y

3.8 axslldp group (LLDP information MIB)

This group provides information about IEEE 802.1AB/D6.0LLDP.

3.8.1 axslldpConfiguration group

(1) ID

```
axslldp OBJECT IDENTIFIER ::= {axsMib 100}

axslldpConfiguration OBJECT IDENTIFIER ::= {axslldp 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.1

axslldpMessageTxInterval OBJECT IDENTIFIER ::= {axslldpConfiguration 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axslldpConfiguration group.

Table 3-20 axslldpConfiguration group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axslldpMessageTxInterval {axslldpConfiguration 1}	Integer32	R/O	[Standard] Sending interval of the LDPDU. Defined by the `lldp interval-time` configuration command (in seconds). The default value is 30 (unit: seconds). [Implementation] Same as the standard.	Y
2	axslldpMessageTxHoldMultiplier {axslldpConfiguration 2}	Integer32	R/O	[Standard] Factor of the period for which the neighboring switch holds LLDP frames sent from the Switch. Actual retention time is the product of the value of axslldpMessageTxInterval and this MIB value. Defined by the `lldp hold-count` configuration command. The default value is 4. [Implementation] Same as the standard.	Y
3	axslldpReinitDelay {axslldpConfiguration 3}	Integer32	R/W	[Standard] Delay time for reinitialization since the time the port state changed to `disable` (in seconds). The default value is 1 (unit: seconds). [Implementation] Not supported.	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	axslldpTxDelay {axslldpConfiguration 4}	Integer32	R/W	[Standard] Delay time until LDPDU is sent after the value of an object of the axslldpLocalSystemData group is changed (in seconds). A recommended value can be calculated using the following expression: axslldpTxDelay = max (1, (0.25 x axslldpMessageTxInterval)) The default value is 8 (unit: seconds). [Implementation] Not supported.	N
5	axslldpPortConfigTable {axslldpConfiguration 6}	SEQUENCE OF axslldpPortConfigEntry	NA	[Standard] Table related to LDPDU transmission. [Implementation] Same as the standard.	Y
6	axslldpPortConfigEntry {axslldpPortConfigTable 1}	axslldpPortConfigEntry	NA	[Standard] Entry (for each port) related to LDPDU transmission. INDEX {axslldpPortConfigPortNum} [Implementation] Same as the standard.	Y
7	axslldpPortConfigPortNum {axslldpPortConfigEntry 2}	Integer32	NA	[Standard] ID to identify ports. Same as ifIndex. [Implementation] Same as the standard.	Y
8	axslldpPortConfigAdminStatus {axslldpPortConfigEntry 3}	INTEGER	R/O	[Standard] Port status of the Switch related to LDPDU transmission and reception. ● txOnly (1) ● rxOnly (2) ● txAndRx (3) ● disabled (4) [Implementation] Only txAndRx (3) and disabled (4) are available.	Y
9	axslldpPortConfigTLVsTxEnable {axslldpPortConfigEntry 4}	BITS	R/O	[Standard] TLV available for transmission related to the corresponding port. ● portDesc (4) ● sysName (5) ● sysDesc (6) ● sysCap (7) [Implementation] Fixed values of portDesc (4), sysName (5), and sysDesc (6).	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
10	axslldpPortConfigRowStatus {axslldpPortConfigEntry 5}	RowStatus	R/O	[Standard] Status of this entry: ● active (1): if axslldpPortConfigAdminStatus is txAndRx ● notReady (3): if axslldpPortConfigAdminStatus is disabled [Implementation] Same as the standard.	Y
11	axslldpConfigManAddrTable {axslldpConfiguration 7}	SEQUENCE OF axslldpConfigManAddrEntry	NA	[Standard] Table related to the ports that send the management address of the Switch. [Implementation] Not supported.	N
12	axslldpConfigManAddrEntry {axslldpConfigManAddrTable 1}	axslldpConfigManAddrEntry	NA	[Standard] Entry constituting the aggregate of ports that send the management address of the Switch [Implementation] Not supported.	N
13	axslldpConfigManAddrPortsTxEnable {axslldpConfigManAddrEntry 1}	OCTET STRING	R/W	[Standard] Bitmap constituting the aggregate of ports that send the management address of the Switch. [Implementation] Not supported.	N

3.8.2 axslldpStats group

(1) ID

```
axslldp OBJECT IDENTIFIER ::= {axsMib 100}

axslldpStats OBJECT IDENTIFIER ::= {axslldp 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.2

axslldpStatsTable OBJECT IDENTIFIER ::= {axslldpStats 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.2.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axslldpStats group.

Table 3-21 axslldpStats group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axslldpStatsTable {axslldpStats 1}	SEQUENCE OF axslldpStatsEntry	NA	[Standard] Table of LLDP statistics. [Implementation] Same as the standard.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
2	axslldpStatsEntry {axslldpStatsTable 1}	axslldpStatsEntry	NA	[Standard] Entry (for each port) related to LLDP statistics. INDEX {axslldpStatsPortNum} [Implementation] Same as the standard.	Y
3	axslldpStatsPortNum {axslldpStatsEntry 2}	Integer32	NA	[Standard] ID to identify ports. Same as ifIndex. [Implementation] Same as the standard. The item is assigned to any port activated with the `lldp enable` configuration command.	Y
4	axslldpStatsOperStatus {axslldpStatsEntry 3}	INTEGER	R/O	[Standard] Status of the port: ● portUp (1) ● portDown (2) [Implementation] Same as the standard.	Y
5	axslldpStatsFramesInErrors {axslldpStatsEntry 4}	Counter32	R/O	[Standard] Number of invalid LDPDUs received by the corresponding port. [Implementation] Same as the standard.	Y
6	axslldpStatsFramesInTotal {axslldpStatsEntry 5}	Counter32	R/O	[Standard] Total number of LDPDUs received by the corresponding port. [Implementation] Same as the standard.	Y
7	axslldpStatsFramesOutTotal {axslldpStatsEntry 6}	Counter32	R/O	[Standard] Total number of LDPDUs sent from the corresponding port. [Implementation] Same as the standard.	Y
8	axslldpStatsTLVsInErrors {axslldpStatsEntry 7}	Counter32	R/O	[Standard] Number of invalid TLVs received by the corresponding port. [Implementation] Same as the standard.	Y
9	axslldpStatsTLVsDiscardedTotal {axslldpStatsEntry 8}	Counter32	R/O	[Standard] Total number of TLVs discarded related to the corresponding port. [Implementation] Same as the standard.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
10	axslldpStatsCounterDiscontinuityTime {axslldpStatsEntry 9}	TimeStamp	R/O	[Standard] Time when the continuity of the statistics counter is lost related to the corresponding port. [Implementation] Not supported.	N

3.8.3 axslldpLocalSystemData group

(1) ID

```
axslldp OBJECT IDENTIFIER ::= {axsMib 100}

axslldpLocalSystemData OBJECT IDENTIFIER ::= {axslldp 3}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.3

axslldpLocChassisType ::= {axslldpLocalSystemData 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.3.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axslldpLocalSystemData group.

Table 3-22 axslldpLocalSystemData group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axslldpLocChassisType {axslldpLocalSystemData 1}	INTEGER	R/O	[Standard] Type of chassis for the Switch: ● entPhysicalAlias (1) ● ifAlias (2) ● portEntPhysicalAlias (3) ● backplaneEntPhysicalAlias (4) ● macAddress (5) ● networkAddress (6) [Implementation] Fixed value of macAddress (5).	Y
2	axslldpLocChassisId {axslldpLocalSystemData 2}	OCTET STRING	R/O	[Standard] Chassis ID (string) of the Switch. [Implementation] MAC address of the switch.	Y
3	axslldpLocSysName {axslldpLocalSystemData 3}	OCTET STRING	R/O	[Standard] System name (string) of the Switch. Same as sysName in the system group. [Implementation] Same as the standard.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	axslldpLocSysDesc {axslldpLocalSystemData 4}	DisplayString	R/O	[Standard] System information (string) for the Switch. Same as sysDesc in the system group. [Implementation] Same as the standard.	Y
5	axslldpLocSysCapSupported {axslldpLocalSystemData 5}	BITS	R/O	[Standard] Bitmap expression for the list of functions supported by the corresponding switch: ● repeater (0) ● bridge (1) ● accessPoint (2) ● router (3) ● telephone (4) ● wirelessStation (5) ● stationOnly (6) [Implementation] Not supported.	N
6	axslldpLocSysCapEnabled {axslldpLocalSystemData 6}	BITS	R/O	[Standard] Bitmap expression for the list of functions running on the corresponding switch: ● repeater (0) ● bridge (1) ● accessPoint (2) ● router (3) ● telephone (4) ● wirelessStation (5) ● stationOnly (6) [Implementation] Not supported.	N
7	axslldpLocPortTable {axslldpLocalSystemData 7}	SEQUENCE OF axslldpLocPortEntry	NA	[Standard] Table related to the ports on the Switch. [Implementation] Same as the standard.	Y
8	axslldpLocPortEntry {axslldpLocPortTable 1}	axslldpLocPortEntry	NA	[Standard] Entry related to each port of the Switch. INDEX {axslldpLocPortNum} [Implementation] Same as the standard.	Y
9	axslldpLocPortNum {axslldpLocPortEntry 1}	Integer32	NA	[Standard] ID to identify ports. Same as ifIndex. [Implementation] Same as the standard. The item is assigned to any port activated with the `lldp enable` configuration command.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
10	axslldpLocPortType {axslldpLocPortEntry 2}	INTEGER	R/O	[Standard] Port type of the corresponding port on the Switch: ● ifAlias (1) ● portEntPhysicalAlias (2) ● backplaneEntPhysicalAlias (3) ● macAddress (4) ● networkAddress (5) ● local (6) [Implementation] Fixed value of macAddress (4).	Y
11	axslldpLocPortId {axslldpLocPortEntry 3}	OCTET STRING	R/O	[Standard] Port ID of the corresponding port on the Switch (string). [Implementation] MAC address of the port.	Y
12	axslldpLocPortDesc {axslldpLocPortEntry 4}	OCTET STRING	R/O	[Standard] Port information of the corresponding port on the Switch (string). Same as ifDescr for a port. [Implementation] Same as the standard.	Y
13	axslldpLocManAddrTable {axslldpLocalSystemData 8}	SEQUENCE OF axslldpLocManAddrEntry	NA	[Standard] Table of the management addresses for the corresponding switch. [Implementation] Not supported.	N
14	axslldpLocManAddrEntry {axslldpLocManAddrTable 1}	axslldpLocManAddrEntry	NA	[Standard] Information on the management address for each chassis. INDEX {axslldpLocManAddrType, axslldpLocManAddr} [Implementation] Not supported.	N
15	axslldpLocManAddrType {axslldpLocManAddrEntry 1}	AddressFamilyNumbers	NA	[Standard] Type of management address. [Implementation] Not supported.	N
16	axslldpLocManAddr {axslldpLocManAddrEntry 2}	OCTET STRING	NA	[Standard] Management address of the switch. [Implementation] Not supported.	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
17	axslldpLocManAddrIfSubtype {axslldpLocManAddrEntry 3}	INTEGER	R/O	[Standard] Type of interface number: ● unknown (1) ● ifIndex (2) ● systemPortNumber (3)} [Implementation] Not supported.	N
18	axslldpLocManAddrIfId {axslldpLocManAddrEntry 4}	OCTET STRING	R/O	[Standard] Interface number corresponding to the management address. [Implementation] Not supported.	N
19	axslldpLocManAddrOID {axslldpLocManAddrEntry 5}	OBJECT IDENTIFIER	R/O	[Standard] Object ID of the hardware or protocol related to the management address. [Implementation] Not supported.	N

3.8.4 axslldpRemoteSystemData group

(1) ID

```
axslldp OBJECT IDENTIFIER ::= {axsMib 100}

axslldpRemoteSystemData OBJECT IDENTIFIER ::= {axslldp 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.4

axslldpRemTable ::= {axslldpRemoteSystemData 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.4.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axslldpRemoteSystemData group.

Table 3-23 axslldpRemoteSystemData group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axslldpRemTable {axslldpRemoteSystemData 1}	SEQUENCE OF axslldpRemEntry	NA	[Standard] Table describing the neighboring switch. [Implementation] Same as the standard.	Y
2	axslldpRemEntry {axslldpRemTable 1}	axslldpRemEntry	NA	[Standard] Entry describing the neighboring switch. INDEX {axslldpRemLocalPortNum ,axslldpRemIndex }#	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
3	axslldpRemTimeMark {axslldpRemEntry 1}	TimeFilter	NA	[Standard] Elapsed time since the information describing the neighboring switch was obtained. [Implementation] Not supported.	N
4	axslldpRemLocalPortNum {axslldpRemEntry 2}	Integer32	R/O	[Standard] ID of the port on the Switch through which the information describing the corresponding neighboring switch is obtained. Same as ifIndex. [Implementation] Same as the standard. The item is assigned to any port activated with the `lldp enable` configuration command.	Y
5	axslldpRemIndex {axslldpRemEntry 3}	Integer32	R/O	[Standard] Index related to the neighboring switch. [Implementation] Same as the standard.	Y
6	axslldpRemRemoteChassisType {axslldpRemEntry 4}	INTEGER	R/O	[Standard] Chassis type of the neighboring switch: ● entPhysicalAlias (1) ● ifAlias (2) ● portEntPhysicalAlias (3) ● backplaneEntPhysicalAlias (4) ● macAddress (5) ● networkAddress (6) [Implementation] Same as the standard.	Y
7	axslldpRemRemoteChassis {axslldpRemEntry 5}	OCTET STRING	R/O	[Standard] Chassis ID of the neighboring switch (string). [Implementation] Same as the standard.	Y
8	axslldpRemRemotePortType {axslldpRemEntry 6}	INTEGER	R/O	[Standard] Port type of the corresponding port on the neighboring switch: ● ifAlias (1) ● portEntPhysicalAlias (2) ● backplaneEntPhysicalAlias (3) ● macAddress (4) ● networkAddress (5) ● local (6) [Implementation] Same as the standard.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
9	axslldpRemRemotePort {axslldpRemEntry 7}	OCTET STRING	R/O	[Standard] Port ID (string) of the corresponding port on the neighboring switch. [Implementation] Same as the standard.	Y
10	axslldpRemPortDesc {axslldpRemEntry 8}	OCTET STRING	R/O	[Standard] Information (string) about the corresponding port on the neighboring switch. [Implementation] Same as the standard.	Y
11	axslldpRemSysName {axslldpRemEntry 9}	OCTET STRING	R/O	[Standard] System name (string) of the neighboring switch. [Implementation] Same as the standard.	Y
12	axslldpRemSysDesc {axslldpRemEntry 10}	OCTET STRING	R/O	[Standard] System (string) information about the neighboring switch. [Implementation] Same as the standard.	Y
13	axslldpRemSysCapSupported {axslldpRemEntry 11}	BITS	R/O	[Standard] Bitmap expression for the list of functions supported by the neighboring switch: ● repeater (0) ● bridge (1) ● accessPoint (2) ● router (3) ● telephone (4) ● wirelessStation (5) ● stationOnly (6) [Implementation] Not supported.	N
14	axslldpRemSysCapEnabled {axslldpRemEntry 12}	BITS	R/O	[Standard] Bitmap expression for the list of functions running on the neighboring switch: ● repeater (0) ● bridge (1) ● accessPoint (2) ● router (3) ● telephone (4) ● wirelessStation (5) ● stationOnly (6) [Implementation] Not supported.	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
15	axslldpRemManAddrTable {axslldpRemoteSystemsData 2}	SEQUENCE OF axslldpRemManAddrEntry	NA	[Standard] Table related to the management addresses of the neighboring switches learned by the corresponding switch. [Implementation] Not supported.	N
16	axslldpRemManAddrEntry {axslldpRemManAddrTable 1}	axslldpRemManAddrEntry	NA	[Standard] Table of the management addresses of the neighboring switch. INDEX {axslldpRemTimeMark, axslldpRemLocalPortNum, axslldpRemIndex, axslldpRemManAddrType, axslldpRemManAddr} [Implementation] Not supported.	N
17	axslldpRemManAddrType {axslldpRemManAddrEntry 1}	AddressFamilyNumbers	NA	[Standard] Type of management address. [Implementation] Not supported.	N
18	axslldpRemManAddr {axslldpRemManAddrEntry 2}	OCTET STRING	NA	[Standard] Management address of the neighboring switch. [Implementation] Not supported.	N
19	axslldpRemManAddrIfSubtype {axslldpRemManAddrEntry 3}	INTEGER	R/O	[Standard] Type of interface number: ● unknown (1) ● ifIndex (2) ● systemPortNumber (3) [Implementation] Not supported.	N
20	axslldpRemManAddrIfId {axslldpRemManAddrEntry 4}	OCTET STRING	R/O	[Standard] Interface number corresponding to the management address of the neighboring switch. [Implementation] Not supported.	N
21	axslldpRemManAddrOID {axslldpRemManAddrEntry 5}	OBJECT IDENTIFIER	R/O	[Standard] Object ID of the hardware or the protocol related to the administration address of the neighboring switch. [Implementation] Not supported.	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
22	axslldpRemOrgDefInfoTable {axslldpRemoteSystemsData 3}	SEQUENCE OF axslldpRemOrgDefInfoEntry	NA	[Standard] Table related to the manufacturer-specific TLVs on the neighboring switch. [Implementation] Same as the standard.	N
23	axslldpRemOrgDefInfoEntry {axslldpRemOrgDefTable 1}	axslldpRemOrgDefInfoEntry	NA	[Standard] Entry related to the manufacturer-specific TLVs on the neighboring switch. INDEX {axslldpRemTimeMark, axslldpRemLocalPortNum, axslldpRemIndex, axslldpRemOrgDefOUI, axslldpRemOrgDefSubtype, axslldpRemOrgDefIndex} [Implementation] Not supported.	N
24	axslldpRemOrgDefInfoOUI {axslldpRemOrgDefEntry 1}	OCTET STRING	NA	[Standard] OUI about the manufacturer-specific TLV on the neighboring switch. [Implementation] Not supported.	N
25	axslldpRemOrgDefInfoSubtype {axslldpRemOrgDefEntry 2}	Integer32	NA	[Standard] Subtype of the manufacturer-specific TLV on the neighboring switch. [Implementation] Not supported.	N
26	axslldpRemOrgDefInfoIndex {axslldpRemOrgDefEntry 3}	Integer32	NA	[Standard] ID of the manufacturer-specific TLV on the neighboring switch. [Implementation] Not supported.	N
27	axslldpRemOrgDefInfo {axslldpRemOrgDefEntry 4}	OCTET STRING	R/O	[Standard] Information on the manufacturer-specific TLV on the neighboring switch. [Implementation] Not supported.	N

#: axslldpRemTimeMark is excluded because it is not supported, even though it is included in the standard.

3.8.5 axslldpRemoteOriginInfoData group

(1) ID

```
axslldp OBJECT IDENTIFIER ::= {axsMib 100}

axslldpRemoteOriginInfoData OBJECT IDENTIFIER ::= {axslldp 20}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.20
```

```
axslldpRemOriginInfoTable OBJECT IDENTIFIER ::=
{axslldpRemoteOriginInfoData 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.100.20.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axslldpRemoteOriginInfoData group.

Table 3-24 axslldpRemoteOriginInfoData group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axslldpRemOriginInfoTable {axslldpRemoteOriginInfoData 1}	SEQUENCE OF axslldpRemOriginInfoTable	NA	Table related to ALAXALA TLVs in the neighboring switch.	Y
2	axslldpRemOriginInfoEntry {axslldpRemOriginInfoTable 1}	axslldpRemOriginInfoEntry	NA	Entry about the neighboring switch. INDEX {axslldpRemOriginInfoLocalPortNum, axslldpRemOriginInfoIndex }	Y
3	axslldpRemOriginInfoPortNum {axslldpRemOriginInfoEntry 1}	Integer32	NA	Port identification index of the port on the Switch that received information on the neighboring switch. Same as ifIndex.	Y
4	axslldpRemOriginInfoIndex {axslldpRemOriginInfoEntry 2}	INTEGER	NA	Index for the neighboring switch.	Y
5	axslldpRemOriginInfoLowerVlanList {axslldpRemOriginInfoEntry 3}	OCTET STRING	R/O	Bitmap expression for valid VLAN IDs (from VLAN 1 to 2047) at the corresponding port on the neighboring switch (the first bit (2^7th bit of the 0th byte) specifies the existence of an untagged definition, and the subsequent bits specify the validity of VLAN 1 to 2047). ● bit is 0: The VLAN is invalid. ● bit is 1: The VLAN is valid.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	axslldpRemOriginInfoHigherVlanList {axslldpRemOriginInfoEntry 4}	OCTET STRING	R/O	Bitmap expression for the valid VLAN IDs (from VLAN 2048 to 4095) on the corresponding port of the neighboring switch (the validity of VLAN 2048 to 4095 is indicated by the sequence that begins from the first bit (2^7th bit of the 0th byte)). ● bit is 0: The VLAN is invalid. ● bit is 1: The VLAN is valid.	Y
7	axslldpRemOriginInfoIPv4Address {axslldpRemOriginInfoEntry 5}	OCTET STRING	R/O	Only if untagged is defined in axslldpRemOriginInfoLowerVlanList: IPv4 address assigned to the corresponding port of the neighboring switch (string) Otherwise: IPv4 address assigned to the VLAN that has the smallest VLAN ID among the valid VLANs on the corresponding port of the neighboring switch (string)	Y
8	axslldpRemOriginInfoIPv4PortType {axslldpRemOriginInfoEntry 6}	INTEGER	R/O	Port type of the corresponding port on the neighboring switch: ● Bridge port (0): Port that does not have an IP address definition, or that is contained by a VLAN that has an assigned IP address ● Router port (1): Any port other than the port described above	Y
9	axslldpRemOriginInfoIPv4VlanId {axslldpRemOriginInfoEntry 7}	INTEGER	R/O	If axslldpRemOriginInfoIPv4PortType is a bridge port: The smallest VLAN ID among the IDs of VLANs that have an assigned IPv4 address If axslldpRemOriginInfoIPv4PortType is a router port: Fixed value of 0.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
10	axslldpRemOriginInfoIPv6Address {axslldpRemOriginInfoEntry 8}	OCTET STRING	R/O	Only if untagged is defined in axslldpRemOriginInfoLowerVlanList: IPv6 address (string) assigned to the corresponding port on the neighboring switch Otherwise: IPv6 address (string) assigned to the VLAN that has the smallest VLAN ID among the valid VLANs on the corresponding port of the neighboring switch	Y
11	axslldpRemOriginInfoIPv6PortType {axslldpRemOriginInfoEntry 9}	INTEGER	R/O	Port type of the corresponding port on the neighboring switch: ● Bridge port (0): Port that does not have an IP address definition, or that is contained by a VLAN that has an assigned IP address ● Router port (1): Any port other than the port described above	Y
12	axslldpRemOriginInfoIPv6VlanId {axslldpRemOriginInfoEntry 10}	INTEGER	R/O	If axslldpRemOriginInfoIPv6PortType is a bridge port: Smallest VLAN ID among the IDs of VLANs that have an assigned IPv6 address If axslldpRemOriginInfoIPv6PortType is a router port: Fixed value of 0.	Y

3.9 axsAxrpMIB group (Ring Protocol information)

3.9.1 axsAxrpGroupTable group

(1) ID

```
axsAxrp OBJECT IDENTIFIER ::= {axsMib 200}

axsAxrpGroupTable OBJECT IDENTIFIER ::= {axsAxrp 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.200.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsAxrpGroupTable group.

Table 3-25 axsAxrpGroupTable group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsAxrpGroupTable {axsAxrp 1}	NOT-ACCESSIBLE	NA	Table containing information on the Ring Protocol group.	Y
2	axsAxrpGroupEntry {axsAxrpGroupTable 1}	NOT-ACCESSIBLE	NA	List of information on the Ring Protocol group. INDEX { axsAxrpGroupRingId }	Y
3	axsAxrpGroupRingId {axsAxrpGroupEntry 1}	INTEGER	NA	Ring ID (1-65535).	Y
4	axsAxrpGroupRowStatus {axsAxrpGroupEntry 2}	RowStatus	R/O	Validity of this entry. Fixed value of Active (1).	Y
5	axsAxrpGroupMode {axsAxrpGroupEntry 3}	INTEGER	R/O	Mode of each Ring ID: ● no-config (1): Configuration command mode is not set up ● master (2): Master node ● transit (3): Transit node	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	axsAxrpGroupRingAttribute {axsAxrpGroupEntry 4}	INTEGER	R/O	In a multi-ring configuration, the attribute of the Switch in the ring without shared link monitoring: ● no-config (1): No attribute ● rift-ring (2): Node contained in a ring without shared link monitoring ● rift-ring-edge 1 (3): Node 1, which is the endmost node on the shared link non-monitoring ring ● rift-ring-edge 2 (4): Node 2, which is the endmost node on the shared link non-monitoring ring	Y
7	axsAxrpGroupMonitoringState {axsAxrpGroupEntry 5}	INTEGER	R/O	Operational and monitoring state of the ring: ● init (1): Initializing ● disable (2): Disabled ● fault monitoring (3): Under fault monitoring ● recovery-monitoring (4): Under recovery monitoring ● flush-monitoring (5): Under monitoring of flush control frames ● not-operating (6): Not operable ● preempt-delay (8): Path switchback is suppressed ● monitoring-recovery (9): Under recovery monitoring (when path switchback suppresion is released)	Y
8	axsAxrpGroupRingport1 {axsAxrpGroupEntry 6}	INTEGER	R/O	ifIndex of ring port 1 (with the smaller ifIndex)[#]	Y
9	axsAxrpGroupRingport1Shared {axsAxrpGroupEntry 7}	INTEGER	R/O	State of sharing of ring port 1.[#] ● no-config (1) ● shared-edge (2) ● shared (3)	Y
10	axsAxrpGroupRingport2 {axsAxrpGroupEntry 8}	INTEGER	R/O	ifIndex of ring port 2 (with the larger ifIndex).[#]	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
11	axsAxrpGroupRingport2Shared {axsAxrpGroupEntry 9}	INTEGER	R/O	State of sharing of ring port 2.# ● no-config (1) ● shared-edge (2) ● shared (3)	Y
12	axsAxrpGroupTransitionToFaultCounts {axsAxrpGroupEntry 10}	Counter	R/O	Number of transitions from the fault monitoring state to the recovery monitoring state.	Y
13	axsAxrpGroupTransitionToNormalCounts {axsAxrpGroupEntry 11}	Counter	R/O	Number of transitions from the recovery monitoring state to the fault monitoring state.	Y
14	axsAxrpGroupLastTransitionTime {axsAxrpGroupEntry 12}	TimeStamp	R/O	Time of the most recent transition from recovery monitoring to fault monitoring, or from fault monitoring to recovery monitoring.	Y
15	axsAxrpGroupMultiFaultDetectionState {axsAxrpGroupEntry 22}	INTEGER	R/O	Multi-fault monitoring of the Ring Protocol: ● not-monitoring (1) ● normal (2) ● fault (3)	Y

#: MIB information for ring port 1 or ring port 2 cannot be obtained unless the ring port is configured. Configure the ring port to obtain the MIB information.

3.9.2 axsAxrpVlanGroupTable group

(1) ID

```
axsAxrp OBJECT IDENTIFIER ::= {axsMib 200}

axsAxrpVlanGroupTable OBJECT IDENTIFIER ::= {axsAxrp 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.200.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsAxrpVlanGroupTable group.

Table 3-26 axsAxrpVlanGroup group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsAxrpVlanGroupTable {axsAxrp 2}	NOT-ACCESSIBLE	NA	Table containing information on the Ring Protocol VLAN groups.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
2	axsAxrpVlanGroupEntry {axsAxrpVlanGroupTable 1}	NOT-ACCESSIBLE	NA	List of ring port information on the VLAN groups. INDEX { axsAxrpVlanGroupRingId, axsAxrpVlanGroupId }	Y
3	axsAxrpVlanGroupRingId {axsAxrpVlanGroupEntry 1}	INTEGER	NA	Ring ID (1-65535).	Y
4	axsAxrpVlanGroupId {axsAxrpVlanGroupEntry 2}	INTEGER	NA	VLAN group ID.	Y
5	axsAxrpVlanGroupRingport1 {axsAxrpVlanGroupEntry 3}	INTEGER	R/O	ifIndex of ring port 1 (with the smaller ifIndex).#	Y
6	axsAxrpVlanGroupRingport1Role {axsAxrpVlanGroupEntry 4}	INTEGER	R/O	Role of ring port 1.# ● primary (1): Primary port of the master node ● secondary (2): Secondary port of the master node ● other (3)	Y
7	axsAxrpVlanGroupRingport1OperState {axsAxrpVlanGroupEntry 5}	INTEGER	R/O	Current state of ring port 1.# ● forwarding (1) ● blocking (2) ● other (3) ● down (4)	Y
8	axsAxrpVlanGroupRingport2 {axsAxrpVlanGroupEntry 6}	INTEGER	R/O	ifIndex of ring port 2 (with the larger ifIndex).#	Y
9	axsAxrpVlanGroupRingport2Role {axsAxrpVlanGroupEntry 7}	INTEGER	R/O	Role of ring port 2.# ● primary (1): Primary port of the master node ● secondary (2): Secondary port of the master node ● other (3)	Y
10	axsAxrpVlanGroupRingport2OperState {axsAxrpVlanGroupEntry 8}	INTEGER	R/O	Current state of ring port 2.# ● forwarding (1) ● blocking (2) ● other (3) ● down (4)	Y

#: MIB information for ring port 1 or ring port 2 cannot be obtained unless the ring port is configured. Configure the ring port to obtain the MIB information.

3.10 axsPconMIB group (power consumption information MIB)

3.10.1 axsPconModuleData group

(1) ID

```
axsPconMIB OBJECT IDENTIFIER ::= {axsMib 300}

axsPconObjects OBJECT IDENTIFIER ::= {axsPconMIB 1}
axsPconModuleData OBJECT IDENTIFIER ::= {axsPconObjects 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.300.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsPconModuleData group.

Table 3-27 axsPconModuleData group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsPconModuleTable { axsPconModuleData 1 }	SEQUENCE OF AxsPconModuleEntry	NA	Table of operating status and power mode of the device.	Y
2	axsPconModuleEntry { axsPconModuleTable 1 }	AxsPconModuleEntry	NA	Device information entry. INDEX{ axsPconModuleIndex }	Y
3	axsPconModuleIndex {axsPconModuleEntry 1}	Integer32	NA	Index information that references each table. ● Entire switch (1000)	Y
4	axsPconModuleType {axsPconModuleEntry 2}	INTEGER	R/O	Type. ● Entire device (1)	Y
5	axsPconModuleSlotNo {axsPconModuleEntry 3}	Integer32	R/O	Position of slots embedded in the entire device. ● Always returns 1 for the entire device.	Y
6	axsPconModuleDescr {axsPconModuleEntry 4}	DisplayString	R/O	Abbreviation of the entire device (up to 16 characters). ● The device model for the entire device (displayed by `show system`). Returns length 0 if no information is available.	Y
7	axsPconModuleStatus {axsPconModuleEntry 5}	INTEGER	R/O	Operating status of the entire device. ● Not implemented (0)	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
8	axsPconModuleMode {axsPconModuleEntry 6}	INTEGER	R/O	Power mode of the entire device. ● normal (1) ● saving (4)	Y

3.10.2 axsPconPowerCon group

(1) ID

```
axsPconMIB OBJECT IDENTIFIER ::= {axsMib 300}

axsPconObjects OBJECT IDENTIFIER ::= {axsPconMIB 1}
axsPconPowerCon OBJECT IDENTIFIER ::= {axsPconObjects 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.1.300.1.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the axsPconPowerCon group.

Table 3-28 axsPconPowerCon group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	axsPconPowerConTable { axsPconPowerCon 1 }	SEQUENCE OF AxsPconPowerConEntry	NA	Table of information about power consumption for the entire device.	Y
2	axsPconPowerConEntry { axsPconPowerConTable 1 }	AxsPconPowerConEntry	NA	Entry of information on power consumption for the entire device. INDEX{ axsPconModuleIndex }	Y
3	axsPconPowerConMaxPower { axsPconPowerConEntry 1 }	Gauge	R/O	Maximum power consumption of the device (unit: 1/100W). ● Fixed value of 0.	Y
4	axsPconPowerConPowerConsumption { axsPconPowerConEntry 2 }	Counter64	R/O	The amount of power consumed by the entire device from startup until now (unit: 1/100Wh).	Y
5	axsPconPowerConPowerMeter { axsPconPowerConEntry 3 }	Gauge	R/O	Current power consumption of the entire device (unit: 1/100W).	Y

3.11 ax2530sSwitch group (system device model information MIB)

(1) ID

```
ax2530sMib     OBJECT IDENTIFIER ::= {axsEx 16}

ax2530sSwitch  OBJECT IDENTIFIER ::= {ax2530sMib 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.1

ax2530sSoftware  OBJECT IDENTIFIER ::= {ax2530sSwitch 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.1.2

ax2530sSystemMsg OBJECT IDENTIFIER ::= {ax2530sSwitch 3}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.1.3

ax2530sSnmpAgent OBJECT IDENTIFIER ::= {ax2530sSwitch 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.1.4

ax2530sLicense   OBJECT IDENTIFIER ::= {ax2530sSwitch 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.1.6
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sSwitch group:

Table 3-29 ax2530sSwitch group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sModelType {ax2530sSwitch 1}	INTEGER	R/O	System device model information (numeric value): ● AX2530S-24T (1600) ● AX2530S-24T4X (1601) ● AX2530S-48T (1603) ● AX2530S-48T2X (1604) ● AX2530S-24S4X (1605) ● AX2530S-24TD (1606) ● AX2530S-48TD (1609) ● AX2530S-24S4XD (1611)	Y
2	ax2530sSoftwareName {ax2530sSoftware 1}	DisplayString	R/O	Software model name in operation. The length is fixed at 0.	Y
3	ax2530sSoftwareAbbreviation {ax2530sSoftware 2}	DisplayString	R/O	Abbreviation of the software in operation.	Y
4	ax2530sSoftwareVersion {ax2530sSoftware 3}	DisplayString	R/O	Version of the software in operation.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
5	ax2530sSystemMsgText {ax2530sSystemMsg 1}	DisplayString	R/O	Latest information in the operation log (string). The latest entry information in the system message log (maximum of 256 characters). For details about log format, refer to *1.2.3 Format of operation logs*, in the manual *Message Log Reference*.	Y
6	ax2530sSystemMsgType {ax2530sSystemMsg 2}	OCTET STRING	R/O	Indicates the event type in one byte: ● An event occurred (01) ● An event is recovered (02)	Y
7	ax2530sSystemMsgTimeStamp {ax2530sSystemMsg 3}	DisplayString	R/O	Time an event occurred (month, day, hours, minutes, seconds), expressed as a 14-byte string in the format of *MM*/*DD hh*:*mm*:*ss*. ● *MM*: month (01–12) ● *DD*: day (01–31) ● *hh*: hour (00–23) ● *mm*: minute (00–59) ● *ss*: second (00–59) A 1-byte space is inserted between *DD* and *hh*.	Y
8	ax2530sSystemMsgLevel {ax2530sSystemMsg 4}	OCTET STRING	R/O	Level of the latest system message log expressed as a 1-byte number: ● Fatal failure (9) ● Severe failure (8) ● Abnormal temperature failure (7) ● Network failure (4) ● Warning (3)	Y
9	ax2530sSystemMsgEventPoint {ax2530sSystemMsg 5}	DisplayString	R/O	String of no more than 8 bytes indicating the event location reported in the system message. For details about event location, see section *1.2.5 (3) Event location* in the manual *Message Log Reference*.	Y
10	ax2530sSystemMsgEventInterfaceID {ax2530sSystemMsg 6}	DisplayString	R/O	String indicating the interface ID in the system message (maximum of 40 characters).	Y
11	ax2530sSystemMsgEventCode {ax2530sSystemMsg 7}	OCTET STRING	R/O	Message ID of the system message, in four bytes (0x00000000–0xFFFFFFFF). Error codes are indicated in four bytes.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
12	ax2530sSystemMsgAdditionalCode {ax2530sSystemMsg 8}	OCTET STRING	R/O	Supplemental system message information, in 6-bytes (0x000000000000–0xFFFFFFFFFFFF). The content of the code is not made public because it is for maintenance purposes.	Y
13	ax2530sSnmpSendReceiveSize {ax2530sSnmpAgent 1}	INTEGER	R/O	Size of the SNMP packets an agent can send and receive (in bytes).	Y
14	ax2530sSnmpReceiveDelay {ax2530sSnmpAgent 2}	INTEGER	R/O	Recommended delaying interval receiving SNMP packets (in milliseconds).	Y
15	ax2530sSnmpContinuousSend {ax2530sSnmpAgent 3}	INTEGER	R/O	Recommended number of sequential sends of SNMP packets.	Y
16	ax2530sSnmpObjectMaxNumber {ax2530sSnmpAgent 4}	INTEGER	R/O	Recommended number of objects in an SNMP packet.	Y
17	ax2530sLicenseNumber {ax2530sLicense 1}	INTEGER	R/O	Number of configured license serial numbers	Y
18	ax2530sLicenseTable {ax2530sLicense 2}	NOT-ACCESSIBLE	NA	Table of license information.	Y
19	ax2530sLicenseEntry {ax2530sLicenseTable 1}	NOT-ACCESSIBLE	NA	License information entry. INDEX {ax2530sLicenseIndex}	Y
20	ax2530sLicenseIndex {ax2530sLicenseEntry 1}	INTEGER	NA	Unique index number assigned to each serial number. Number in the range 1 through ax2530sLicenseNumber.	Y
21	ax2530sLicenseSerialNumber {ax2530sLicenseEntry 2}	DisplayString	R/O	Serial number.	Y
22	ax2530sLicenseOptionNumber {ax2530sLicenseEntry 3}	INTEGER	R/O	Number of licenses related to a serial number.	Y
23	ax2530sLicenseOptionTable {ax2530sLicense 3}	NOT-ACCESSIBLE	NA	Table of license information related to a serial number.	Y
24	ax2530sLicenseOptionEntry {ax2530sLicenseOptionTable 1}	NOT-ACCESSIBLE	NA	Entry of license information related to a serial number. INDEX {ax2530sLicenseOptionIndex ax2530sLicenseOptionNumberIndex }	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
25	ax2530sLicenseOptionIndex {ax2530sLicenseOptionEntry 1}	INTEGER	NA	Unique index number assigned to each serial number. Same as ax2530sLicenseIndex	Y
26	ax2530sLicenseOptionNumberIndex {ax2530sLicenseOptionEntry 2}	INTEGER	NA	Index number of the license information related to a serial number. Number in the range 1 through ax2530sLicenseOptionNumber	Y
27	ax2530sLicenseOptionSoftwareName {ax2530sLicenseOptionEntry 3}	DisplayString	R/O	Software model name of the license information related to a serial number.	Y
28	ax2530sLicenseOptionSoftwareAbbreviation {ax2530sLicenseOptionEntry 4}	DisplayString	R/O	Abbreviation of the software of the license information related to a serial number.	Y

3.12 ax2530sDevice group (system device chassis information MIB)

3.12.1 ax2530sChassis group implementation specifications (chassis information)

(1) ID

```
ax2530sDevice  OBJECT IDENTIFIER ::= {ax2530sMib 2}
ax2530sChassis          OBJECT IDENTIFIER ::= {ax2530sDevice 1}

ax2530sChassisMaxNumber OBJECT IDENTIFIER ::= {ax2530sChassis 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.1

ax2530sChassisTable     OBJECT IDENTIFIER ::= {ax2530sChassis 2}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.2
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sChassis group:

Table 3-30 ax2530sChassis group implementation specifications (chassis information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sChassisMaxNumber {ax2530sChassis 1}	INTEGER	R/O	Maximum number of cluster chassis that can be connected to the Switch. For AX2530S: Fixed value of 1	Y
2	ax2530sChassisTable {ax2530sChassis 2}	NOT-ACCESSIBLE	NA	Table of the chassis information.	Y
3	ax2530sChassisEntry {ax2530sChassisTable 1}	NOT-ACCESSIBLE	NA	Entry of Information on a specific chassis. INDEX {ax2530sChassisIndex}	Y
4	ax2530sChassisIndex {ax2530sChassisEntry 1}	NOT-ACCESSIBLE	NA	Number to identify ax2530sChassisEntry. Fixed value of 1.	Y
5	ax2530sChassisType {ax2530sChassisEntry 2}	INTEGER	R/O	Chassis type: ● AX2530S-24T (1600) ● AX2530S-24T4X (1601) ● AX2530S-48T (1603) ● AX2530S-48T2X (1604) ● AX2530S-24S4X (1605) ● AX2530S-24TD (1606) ● AX2530S-48TD (1609) ● AX2530S-24S4XD (1611)	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	ax2530sChassisStatus {ax2530sChassisEntry 3}	INTEGER	R/O	Current status of the chassis. Fixed value of (2) during operation.	Y
7	ax2530sStsLedStatus {ax2530sChassisEntry 4}	INTEGER	R/O	Status of the ST1 LED on the Switch. For the conditions, refer to the *Hardware Instruction Manual*. ● Blinking green (1) ● Stable green (2) ● Blinking red (3) ● Stable red (4) ● Not lit (6) ● Stable orange (8)	Y
8	ax2530sCpuName {ax2530sChassisEntry 5}	DisplayString	R/O	CPU name (maximum of 16 characters).	N
9	ax2530sCpuClock {ax2530sChassisEntry 6}	INTEGER	R/O	CPU clock rate (in MHz).	N
10	ax2530sMemoryTotalSize {ax2530sChassisEntry 7}	INTEGER	R/O	Amount of installed memory (in KB).	Y
11	ax2530sMemoryUsedSize {ax2530sChassisEntry 8}	INTEGER	R/O	Amount of used memory (in KB).	Y
12	ax2530sMemoryFreeSize {ax2530sChassisEntry 9}	INTEGER	R/O	Amount of unused memory (in KB).	Y
13	ax2530sRomVersion {ax2530sChassisEntry 10}	DisplayString	R/O	Version of the installed ROM (string).	N
14	ax2530sCpuLoad1m {ax2530sChassisEntry 11}	INTEGER	R/O	Percentage of CPU usage over a one-minute period (0–100).	Y
15	ax2530sFlashTotalSize {ax2530sChassisEntry 12}	INTEGER	R/O	Sum of the amount of used and unused memory in the file system of the embedded flash memory (in KB).	N
16	ax2530sFlashUsedSize {ax2530sChassisEntry 13}	INTEGER	R/O	Amount of used memory in the file system of the embedded flash memory (in KB).	N
17	ax2530sFlashFreeSize {ax2530sChassisEntry 14}	INTEGER	R/O	Amount of unused memory in the file system of the embedded flash memory (in KB).	N

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
18	ax2530sSdCardStatus {ax2530sChassisEntry 15}	INTEGER	R/O	Status of the connection with the memory card. ● Connected (2) ● Not connected (32)	N
19	ax2530sSdCardTotalSize {ax2530sChassisEntry 16}	INTEGER	R/O	Total capacity of the memory card (in KB).	N
20	ax2530sSdCardUsedSize {ax2530sChassisEntry 17}	INTEGER	R/O	Used capacity of the memory card (in KB).	N
21	ax2530sSdCardFreeSize {ax2530sChassisEntry 18}	INTEGER	R/O	Amount of unused memory card capacity (in KB).	N
22	ax2530sPhysLineNumber {ax2530sChassisEntry 19}	INTEGER	R/O	Number of ports available for connection to this chassis.	Y
23	ax2530sTemperatureStatusNumber {ax2530sChassisEntry 20}	INTEGER	R/O	Maximum number of temperature monitoring points in this chassis. For AX2530S: Fixed value of 1.	Y
24	ax2530sPowerUnitNumber {ax2530sChassisEntry 21}	INTEGER	R/O	Number of power supplies available in this chassis. For AX2530S: Fixed value of 1.	Y
25	ax2530sRedundantPsNumber {ax2530sChassisEntry 22}	INTEGER	R/O	Number of external power units that can be installed in this chassis. For AX2530S: Fixed value of 1.	Y
26	ax2530sFanNumber {ax2530sChassisEntry 23}	INTEGER	R/O	Number of main fans on the chassis. ● For AX2530S-24T: 0 ● For AX2530S-24T4X: 2 ● For AX2530S-48T: 2 ● For AX2530S-48T2X: 2 ● For AX2530S-24S4X: 4 ● For AX2530S-24TD: 0 ● For AX2530S-48TD: 2 ● For AX2530S-24S4XD: 4	Y
27	ax2530sTotalAccumRunTime {ax2530sChassisEntry 24}	INTEGER	R/O	Total run time of the Switch since startup.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
28	ax2530sCriticalAccumRunTime{ax2530sChassisEntry 25}	INTEGER	R/O	● For AX2530S-24T/AX2530S-24TD: Total time the Switch ran in an environment that is above 45 degrees Celcius. ● For Others : Total time the Switch ran in an environment that is above 50 degrees Celcius.	Y
29	ax2530sSts2LedStatus {ax2530sChassisEntry 26}	INTEGER	R/O	Status of ST2 LED on the Switch. For lighting conditions, refer to the *Hardware Instruction Manual*. ● Blinking green (1) ● Stable green (2) ● Not lit (6) ● Stable orange (8)	Y

3.12.2 ax2530sChassis group implementation specifications (temperature information)

(1) ID

```
ax2530sChassis          OBJECT IDENTIFIER ::= {ax2530sDevice 1}

ax2530sTemperatureStatusTable OBJECT IDENTIFIER ::= {ax2530sChassis 3}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sChassis group (temperature information):

Table 3-31 ax2530sChassis group implementation specifications (temperature information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sTemperatureStatusTable {ax2530sChassis 3}	NOT-ACCESSIBLE	NA	Table of temperature status.	Y
2	ax2530sTemperatureStatusEntry {ax2530sTemperatureStatusTable 1}	NOT-ACCESSIBLE	NA	Entry of temperature status. INDEX {ax2530sChassisIndex,ax2530sTemperatureStatusIndex}	Y
3	ax2530sTemperatureStatusIndex {ax2530sTemperatureStatusEntry 1}	NOT-ACCESSIBLE	NA	Unique index number assigned to each temperature monitoring point.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	ax2530sTemperatureStatusDescr {ax2530sTemperatureStatusEntry 2}	DisplayString	R/O	Description of the temperature monitoring point. ● Main board Temperature: The temperature of the main board	Y
5	ax2530sTemperatureStatusValue {ax2530sTemperatureStatusEntry 3}	Integer32	R/O	Current temperature of this monitoring point. The temperature is 0 degrees Celsius for 60 minutes after the device is activated.	Y
6	ax2530sTemperatureThreshold {ax2530sTemperatureStatusEntry 4}	Integer32	R/O	Temperature on this monitoring point when a warning is issued to the Switch.	Y
7	ax2530sTemperatureState {ax2530sTemperatureStatusEntry 5}	INTEGER	R/O	Current temperature status of this monitoring point. ● Normal (1) ● Caution (2) ● Abnormal (4)	Y

3.12.3 ax2530sChassis group implementation specifications (power supply information)

(1) ID

```
ax2530sChassis        OBJECT IDENTIFIER ::= {ax2530sDevice 1}

ax2530sPowerUnitTable OBJECT IDENTIFIER ::= {ax2530sChassis 4}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.4
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sChassis group (power supply information):

Table 3-32 ax2530sChassis group implementation specifications (power supply information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sPowerUnitTable {ax2530sChassis 4}	NOT-ACCESSIBLE	NA	Table of power supply information.	Y
2	ax2530sPowerUnitEntry {ax2530sPowerUnitTable 1}	NOT-ACCESSIBLE	NA	Entry of power supply information. INDEX {ax2530sChassisIndex,ax2530sPowerUnitIndex}	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
3	ax2530sPowerUnitIndex {ax2530sPowerUnitEntry 1}	NOT-ACCESSIBLE	NA	Index indicating the position of the power supply. Number in the range 1 through ax2530sPowerUnitNumber. ● For AX2530S : 1	Y
4	ax2530sPowerConnectStatus {ax2530sPowerUnitEntry 2}	INTEGER	R/O	Installation status of the power supplies: ● Installed (2) ● Not installed (32) Fixed value of 2.	Y
5	ax2530sPowerSupplyStatus {ax2530sPowerUnitEntry 3}	INTEGER	R/O	Power status: ● In operation (2) ● In failure (4)	Y

3.12.4 ax2530sChassis group implementation specifications (fan information)

(1) ID

```
ax2530sChassis   OBJECT IDENTIFIER ::= {ax2530sDevice 1}

ax2530sFanTable  OBJECT IDENTIFIER ::= {ax2530sChassis 5}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.5
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sChassis group (fan information):

Table 3-33 ax2530sChassis group implementation specifications (fan information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sFanTable {ax2530sChassis 5}	NOT-ACCESSIBLE	NA	Table of fan information.	Y
2	ax2530sFanEntry {ax2530sFanTable 1}	NOT-ACCESSIBLE	NA	Entry of fan information. INDEX {ax2530sChassisIndex, ax2530sFanIndex}	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
3	ax2530sFanIndex {ax2530sFanEntry 1}	NOT-ACCESSIBLE	NA	Index indicating the position of the main fan. Number in the range 1 through ax2530sFanNumber: ● For AX2530S-24T: None ● For AX2530S-24T4X: 1 to 2 ● For AX2530S-48T: 1 to 2 ● For AX2530S-48T2X: 1 to 2 ● For AX2530S-24S4X: 1 to 4 ● For AX2530S-24TD: None ● For AX2530S-48TD: 1 to 2 ● For AX2530S-24S4XD: 1 to 4	Y
4	ax2530sFanStatus {ax2530sFanEntry 2}	INTEGER	R/O	Main fan status: ● In operation (2) ● In failure (4) ● Stopped (5)	Y

3.12.5 ax2530sChassis group implementation specifications (external power information)

(1) ID

```
ax2530sChassis          OBJECT IDENTIFIER ::= {ax2530sDevice 1}

ax2530sRedundantPsTable OBJECT IDENTIFIER ::= {ax2530sChassis 6}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.1.6
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sChassis group (external power information):

Table 3-34 ax2530sChassis group implementation specifications (external power information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sRedundantPsTable {ax2530sChassis 6}	NOT-ACCESSIBLE	NA	Table of the external power supply unit infomation	Y
2	ax2530sRedundantPsEntry {ax2530sRedundantPsTable 1}	NOT-ACCESSIBLE	NA	Entry of the external power supply unit infomation INDEX {ax2530sChassisIndex, ax2530sRedundantPsIndex }	Y
3	ax2530sRedundantPsIndex {ax2530sRedundantPsEntry 1}	NOT-ACCESSIBLE	NA	Index indicating the position of the external power supply unit. Number in the range 1 through ax2530sRedundantPsNumber: ● For AX2530S: 1	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	ax2530sRedundantPsConnectStatus {ax2530sRedundantPsEntry 2}	INTEGER	R/O	Installation status of the external power supply unit: ● Installed (2) ● Not installed (32)	Y
5	ax2530sRedundantPsStatus {ax2530sRedundantPsEntry 3}	INTEGER	R/O	Status of the external power supply unit: ● In operation (2) ● In failure (4) Returns –1 when no external power is implemented.	Y

3.12.6 ax2530sPhysLine group implementation specifications ((physical) line information)

(1) ID

```
ax2530sPhysLine      OBJECT IDENTIFIER ::= {ax2530sDevice 2}

ax2530sPhysLineTable OBJECT IDENTIFIER ::= {ax2530sPhysLine 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.2.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sPhysLine group ((physical) line information):

Table 3-35 ax2530sPhysLine group implementation specifications ((physical) line information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sPhysLineTable {ax2530sPhysLine 1}	NOT-ACCESSIBLE	NA	Table of physical line information.	Y
2	ax2530sPhysLineEntry {ax2530sPhysLineTable 1}	NOT-ACCESSIBLE	NA	Physical line information entry. INDEX {ax2530sChassisIndex, ax2530sPhysLineIndex }	Y
3	ax2530sPhysLineIndex {ax2530sPhysLineEntry 1}	NOT-ACCESSIBLE	NA	Physical line number. Number in the range 1 through ax2530sPhysLineNumber.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
4	ax2530sPhysLineConnectorType {ax2530sPhysLineEntry 2}	INTEGER	R/O	Type of interface on the interchangeable transceiver. ● other (1) ● type100BASE-FX (201) ● type1000BASE-LX (301) ● type1000BASE-SX (302) ● type1000BASE-LH (303) ● type1000BASE-BX10-D (304) ● type1000BASE-BX10-U (305) ● type1000BASE-BX40-D (306) ● type1000BASE-BX40-U (307) ● type1000BASE-SX2 (308) ● type1000BASE-UTP (309) ● type1000BASE-LHB (310) ● type10GBASE-SR (401) ● type10GBASE-LR (402) ● type10GBASE-ER (403) ● type10GBASE-CU1M (405) ● type10GBASE-CU3M (406) ● type10GBASE-CU5M (407) ● type10GBASE-CU30CM (408) Returns other (1) when: ● The tranceiver type is unknown or the transceiver is not interchangeable.	Y
5	ax2530sPhysLineOperStatus {ax2530sPhysLineEntry 3}	INTEGER	R/O	Status of the physical line: ● other (1) ● In operation (2) ● initializing (3) ● failure (4) ● operation stopped for configuration purpose (6) ● In operation (now in line fault) (8) ● line testing (9) ● stopped (InActive) (20)	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
6	ax2530sPhysLineIfIndex Number {ax2530sPhysLineEntry 4}	INTEGER	R/O	Number of ifIndex objects in the interface.	Y
7	ax2530sPhysLineTransceiverStatus {ax2530sPhysLineEntry 5}	INTEGER	R/O	Type and installation status of the interchangeable transceiver. ● Not interchangeable (1) ● SFP+/SFP installed (20) ● SFP+/SFP not installed (21) ● Unsupported SFP+/SFP installed (22) ● SFP+/SFP status unknown (23)	Y

3.12.7 ax2530sDeviceError group implementation specifications (device failure information)

(1) ID

```
ax2530sDevice  OBJECT IDENTIFIER ::= {ax2530sMib 2}

ax2530sDeviceError  OBJECT IDENTIFIER ::= {ax2530sDevice 3}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.2.3
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sDeviceError group (device failure information):

Table 3-36 ax2530sDeviceError group implementation specifications (device failure information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sMemoryError {ax2530sDeviceError 1}	Integer32	NA	Details of the memory error other (0) Parity Error (1)	Y

3.13 ax2530sAuth group (authentication information)

(1) ID

```
ax2530sAuth       OBJECT IDENTIFIER ::= {ax2530sMib 10}

ax2530sAuthInfo   OBJECT IDENTIFIER ::= {ax2530sAuth 1}
Object ID value: 1.3.6.1.4.1.21839.2.2.16.10.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sAuth group (common authentication information):

Table 3-37 ax2530sAuth group implementation specifications (common authentication information)

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sAuthSysName {ax2530sAuthInfo 1}	OCTET STRING	NA	Host name of the Switch. (same as sysName in system group)	Y
2	ax2530sAuthIfIndex {ax2530sAuthInfo 2}	Integer32	NA	Port number of the authenticated terminal. (same as ifIndex in interfaces group)	Y
3	ax2530sAuthSupplicantMac {ax2530sAuthInfo 3}	MacAddress	NA	MAC address of the authenticated terminal.	Y
4	ax2530sAuthMessage {ax2530sAuthInfo 4}	DisplayString	NA	Message (maximum of 280 characters).	Y

3.14 ax2530sSml group (SML information) [OS-L2A]

(1) ID

```
ax2530sSml          OBJECT IDENTIFIER ::= {ax2530sMib 20}
ax2530sSmlInfo      OBJECT IDENTIFIER ::= {ax2530sSml 1}

Object ID value: 1.3.6.1.4.1.21839.2.2.16.20.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the ax2530sSml group.

Table 3-38 ax2530sSml group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	ax2530sSmlDomain {ax2530sSmlInfo 1}	Integer32	R/O	SML domain ID	Y
2	ax2530sSmlId {ax2530sSmlInfo 2}	Integer32	R/O	Device ID used by SML functionality.	Y

3.15 sFlow group (InMon private MIB)

(1) ID

```
InMon           OBJECT IDENTIFIER ::= {enterprises 4300}
sFlowMIB        OBJECT IDENTIFIER ::= {InMon 1}

sFlowAgent      OBJECT IDENTIFIER ::= {sFlowMIB 1}
Object ID value: 1.3.6.1.4.1.4300.1.1
```

(2) Implementation specifications

The following table shows the implementation specifications for the sFlow group.

Table 3-39 sFlow group implementation specifications

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
1	sFlowVersion {sFlowAgent 1}	SnmpAdminString	R/O	[Standard] Version of sFlow. A string in the following format: *<MIB Version>;<Organization>;<Software Revision>*. [Implementation] Same as the standard. Fixed value of 1.2;;.	Y
2	sFlowAgentAddressType {sFlowAgent 2}	InetAddressType	R/O	[Standard] Type of IP address of the agent that gathers statistics: ● ipv4 (1) ● ipv6 (2) [Implementation] Same as the standard.	Y#1#2
3	sFlowAgentAddress {sFlowAgent 3}	InetAddress	R/O	[Standard] IP address (IPv4, IPv6) of the agent that gathers statistics. [Implementation] Same as the standard.	Y#1#2
4	sFlowTable {sFlowAgent 4}	NOT-ACCESSIBLE	NA	[Standard] Table containing information on sFlow packets. [Implementation] Same as the standard.	Y
5	sFlowEntry {sFlowTable 1}	NOT-ACCESSIBLE	NA	[Standard] List of information on sFlow packets. INDEX { sFlowDataSource } [Implementation] Same as the standard.	Y
6	sFlowDataSource {sFlowEntry 1}	OBJECT IDENTIFIER	R/O	[Standard] Source of sFlow packet information. Object ID for the ifIndex of the Ethernet port. [Implementation] Same as the standard.	Y

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
7	sFlowOwner {sFlowEntry 2}	OwnerString	R/NW	[Standard] Owner using the sFlow sampler. [Implementation] Same as the standard. The fixed value is blank, and is Read_Only.	M
8	sFlowTimeout {sFlowEntry 3}	Integer32	R/NW	[Standard] Time until the sFlow sampler stops from the time it is released (in seconds). [Implementation] Same as the standard. Fixed value of 0, and is Read_Only.	M
9	sFlowPacketSamplingRate {sFlowEntry 4}	Integer32	R/NW	[Standard] Interval of sFlow packet sampling. [Implementation] Same as the standard. Read_Only.	Y#1#4
10	sFlowCounterSamplingInterval {sFlowEntry 5}	Integer32	R/NW	[Standard] Sending interval of counter samples to the collector. [Implementation] Same as the standard. Read_Only.	Y#1#4
11	sFlowMaximumHeaderSize {sFlowEntry 6}	Integer32	R/NW	[Standard] Maximum length of the copied object from the head of the sample packet. [Implementation] Same as the standard. Read_Only.	Y#1#4
12	sFlowMaximumDatagramSize {sFlowEntry 7}	Integer32	R/NW	[Standard] Maximum number of bytes of data that can be sent as sFlow packets. [Implementation] Same as the standard. Read_Only.	Y#1#4
13	sFlowCollectorAddressType {sFlowEntry 8}	InetAddressType	R/NW	[Standard] Type of IP address of the collector. ● ipv4 (1) ● ipv6 (2) [Implementation] Same as the standard. Read_Only.	Y#1#3#4
14	sFlowCollectorAddress {sFlowEntry 9}	InetAddress	R/NW	[Standard] IP address of the collector. [Implementation] Same as the standard. Read_Only.	Y#1#3#4
15	sFlowCollectorPort {sFlowEntry 10}	Integer32	R/NW	[Standard] Destination port of the sFlow packets. [Implementation] Same as the standard. Read_Only.	Y#1#4

#	Object identifier	SYNTAX	Access	Implementation specifications	Implemented
16	sFlowDatagramVersion {sFlowEntry 11}	Integer32	R/NW	[Standard] Version of the sFlow packets. Versions 2 and 4 are supported. [Implementation] Same as the standard. Read_Only.	Y#1#4

#1: The entry takes the value defined by an sFlow statistics configuration command.

#2: Returns the IPv4 address if both an IPv4 address and an IPv6 address are set.

#3: If multiple collectors are set up, the entry takes the value of the collector with the lowest IP address.

#4: Ports that have no sFlow set are fixed to a value such as 0 according to the RFC.

4. Supported MIB Traps

This chapter describes the supported MIB traps.

4.1 Types and issuance timings of supported traps

The following table lists the supported traps and the timing at which they are issued.

Table 4-1 Supported trap types and the timing at which they are issued

#	Trap type	Meaning	Issued when	Implemented
1	coldStart	An object in a system re-initialization might have changed.	This trap is issued when any of the following occurs:# 1. The a device starts. 2. An agent is re-initialized with a configuration change (changed interface or protocol). 3. The `set clock` command was used to change the time.	Y
2	warmStart	The object in a system re-initialization is not changed.	This trap is issued when the SNMP configuration is changed.	Y
3	linkDown	Line failure detection	This trap is issued when the operating state of an interface is changed from ACTIVE (communication enabled) to DISABLE (communication disabled).	Y
4	linkUp	Line failure recovery	This trap is issued when the operating state of an interface is changed from DISABLE (communication disabled) to ACTIVE (communication enabled).	Y
5	authenticationFailure	Confirmation error	This trap is issued when an SNMP packet is received from an invalid community (that is, an authentication error occurs).	Y
6	risingAlarm	A value exceeded an upper threshold.	This trap is issued when a value exceeds the upper threshold of an RMON alarm.	Y
7	fallingAlarm	A value fell below a lower threshold.	This trap is issued when a value falls below the lower threshold of an RMON alarm.	Y
8	ax2530sSystemMsgTrap	System message output	This trap is issued when a system message is output.	N
9	ax2530sTemperatureTrap	Transition of temperature state	This trap is issued when the monitored temperature changed to a normal, caution, or abnormal state.	Y
10	ax2530sAxrpStateTransitionTrap	Transition of Ring Protocol ring state	This trap is issued when the ring state (failure monitoring and recovery monitoring) of the Ring Protocol changes.	Y
11	ax2530sAxrpMultiFaultDetectionStartTrap	Multi-fault monitoring start of the Ring Protocol	This trap is issued when the multi-fault monitoring for the Ring Protocol is started.	Y

#	Trap type	Meaning	Issued when	Implemented
12	ax2530sAxrpMultiFaultDetectionStateTransitionTrap	Transition of multi-fault monitoring for the Ring Protocol	This trap is issued when the multi-fault monitoring state (multi-fault monitoring and recovery monitoring) of the Ring Protocol changes.	Y
13	ax2530sAirFanStopTrap	A fan is out of order.	This trap is issued when a problem with a fan is detected.	Y
14	ax2530sPowerSupplyFailureTrap	Power is out of order.	● When an abnormality occurs in any of the installed power supplies. ● When a switch loses power.	Y
15	ax2530sLoginSuccessTrap	A device user has logged in successfully.	This trap is issued when a user succeeds in logging in by using the console, Telnet, or FTP.	Y
16	ax2530sLoginFailureTrap	A device user has failed to log in.	● This trap is sent each time user authentication fails during an attempt to log in via console, Telnet, or FTP. ● This trap is not sent when a connection is lost due to a remote access restriction, a timeout occurs at the `Login:` or `Password:` prompt, or a forced disconnection occurs. (In addition, this trap is not sent when only the **Enter** key is pressed at the `Login:` prompt.)	Y
17	ax2530sLogoutTrap	A device user has logged out.	This trap is issued when a user succeeds in logging out using the console, Telnet, or FTP.	Y
18	ax2530sMemoryUsageTrap	Running short of available memory.	This trap is issued when usable memory falls below the lower threshold.	N
19	axsOadpNeighborCachelastChangeTrap	Information on an OADP neighboring node has been updated.	This trap is issued when information on an OADP neighboring node is updated.	N
20	ax2530sFrameErrorReceiveTrap	A frame reception error occurred.	This trap is issued when a frame reception error occurs.	N
21	ax2530sFrameErrorSendTrap	A frame transmission error occurred.	This trap is issued when a frame transmission error occurs.	N
22	ax2530sBroadcastStormDetectTrap	Storm detection	This trap is issued when a broadcast storm is detected and the port is not inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
23	ax2530sMulticastStormDetectTrap	Storm detection	This trap is issued when a multicast storm is detected and the port is not inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y

#	Trap type	Meaning	Issued when	Implemented
24	ax2530sUnicastStormDetectTrap	Storm detection	This trap is issued when a unicast storm is detected and the port is not inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
25	ax2530sBroadcastStormPortInactivateTrap	A port is inactivated by storm detection.	This trap is issued when a broadcast storm is detected and the port is inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
26	ax2530sMulticastStormPortInactivateTrap	A port is inactivated by storm detection.	This trap is issued when a multicast storm is detected and the port is inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
27	ax2530sUnicastStormPortInactivateTrap	A port is inactivated by storm detection.	This trap is issued when a unicast storm is detected and the port is inactivated. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
28	ax2530sBroadcastStormRecoverTrap	Storm termination	This trap is issued when the termination of a broadcast storm is detected. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
29	ax2530sMulticastStormRecoverTrap	Storm termination	This trap is issued when the termination of a multicast storm is detected. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
30	ax2530sUnicastStormRecoverTrap	Storm termination	This trap is issued when the termination of a unicast storm is detected. The setting for detection of this error can be configured using the `storm-control` configuration command.	Y
31	ax2530sEfmoamUdldPortInactivateTrap	A port is inactivated by detection of a one-way link failure.	This trap is issued when a one-way link failure is detected and a port is inactivated. The setting for detection of this error can be configured using the `efmoam active` configuration command.	Y
32	ax2530sEfmoamLoopDetectPortInactivateTrap	A port is inactivated by loop detection.	This trap is issued when the loop state is detected and a port is inactivated. The setting for detection of this error can be configured using the `efmoam active` configuration command.	N
33	pethPsePortOnOffNotification	PD power status	This trap is issued when the status of the power supply to the PD changes between ON and OFF.	N
34	pethMainPowerUsageOnNotification	Power overused.	This trap is issued when the total power consumption of a device exceeds the threshold value.	N

#	Trap type	Meaning	Issued when	Implemented
35	pethMainPowerUsageOffNotification	Power underused.	This trap is issued when the total power consumption of a device falls below the threshold value.	N
36	ax2530sDot1xFailureTrap	IEEE 802.1X authentication failure	This trap is issued when, during IEEE 802.1X authentication, authentication fails.	Y
37	ax2530sDot1xEventTrap	IEEE 802.1X authentication event other than authentication failure	This trap is issued when, during IEEE 802.1X authentication, a login or logout event other than an authentication failure occurs.	Y
38	ax2530sWauthFailureTrap	Web authentication failure	This trap is issued when, during Web authentication, an authentication fails.	Y
39	ax2530sWauthEventTrap	Web authentication event other than authentication failure	This trap is issued when, during Web authentication, a login or logout event other than an authentication failure occurs.	Y
40	ax2530sMauthFailureTrap	MAC authentication failure	This trap is issued when, during MAC authentication, an authentication fails.	Y
41	ax2530sMauthEventTrap	MAC authentication event other than authentication failure	This trap is issued when, during MAC authentication, a login or logout event other than an authentication failure occurs.	Y
42	ax2530sDot1xSystemTrap	Specific SYSTEM notification during IEEE 802.1X authentication	This trap is issued for a specific SYSTEM account log notification during IEEE 802.1X authentication.	Y
43	ax2530sWauthSystemTrap	Specific SYSTEM notification during Web authentication	This trap is issued for a specific SYSTEM account log notification during Web authentication.	Y
44	ax2530sMauthSystemTrap	Specific SYSTEM notification during MAC authentication	This trap is issued for a specific SYSTEM account log notification during MAC authentication.	Y
45	ax2530sL2ldLinkDown	Transition of a line to the communication disabled state by L2 loop detection	This trap is issued when an L2 loop is detected and the operating state of the interface has changed from active (communication possible) to disable (communication impossible).	Y

#	Trap type	Meaning	Issued when	Implemented
46	ax2530sL2ldLinkUp	Transition of a line to the communication enabled state by the automatic recovery functionality of L2 loop detection	This trap is issued when the automatic recovery functionality of L2 loop detection causes the operating state of an interface to change from disable (communication not possible) to active (communication possible).	Y
47	ax2530sL2ldLoopDetection	L2 loop detection	A L2 loop is detected. This trap is issued once every 60 seconds while the L2 loop continues.	Y
48	ax2530sUlrChangeSecondary	Line switchover by uplink redundancy (from the primary to secondary)	This trap is issued when uplink redundancy has completed a switchover from the primary to the secondary line.	Y
49	ax2530sUlrChangePrimary	Line switchover by uplink redundancy (from the secondary to primary)	This trap is issued when uplink redundancy has completed a switchover from the secondary to the primary line.	Y
50	dot1agCfmFaultAlarm	CFM failure detected.	This trap is issued when a CFM failure is detected.	Y
51	ax2530sDeviceErrorTrap	Switch failure detected.	This trap is issued when a switch failure is detected.	Y
52	ax2530sSmlStatusFull [OS-L2A]	SML status (full)	This trap is issued when the SML is in the full status.	Y
53	ax2530sSmlStatusStandalone [OS-L2A]	SML status (standalone)	This trap is issued when the SML is in the standalone status.	Y
54	ax2530sSmlStatusConflict [OS-L2A]	SML status (confilct)	This trap is issued when the SML is in the conflict status.	Y
55	ax2530sSmlPeerlinkNormal [OS-L2A]	Peer link port is up (all ports)	This trap is issued when all ports specified in a peer link are in the link-up status.	Y
56	ax2530sSmlPeerlinkFailure [OS-L2A]	Peer link port is down	This trap is issued when some peer link ports are down.	Y
57	ax2530sSmlPeerlinkRecovery [OS-L2A]	Peer link port is up	This trap is issued when some peer link ports are up.	Y
58	ax2530sSmlPeerlinkDisconnect [OS-L2A]	Peer link port is down (all ports)	This trap is issued when all ports specified in a peer link are in the link-down status.	Y

#

A coldStart trap is sent 5 minutes after the switch starts. If a different trap event occurs before the coldStart trap is sent, the trap is discarded.

(Legend)

Y: Indicates a trap that is supported in (responds to) the Switch.

N: Indicates a trap that is not supported in (does not responds to) the Switch.

n/a: Not applicable

4.2 Supported traps of PDU parameters

Table 4-2 Supported Trap-PDU parameters (for SNMPv1) shows the supported Trap-PDU parameters for SNMPv1. *Table 4-3 Supported Trap-PDU parameters (for SNMPv2C/SNMPv3)* shows the supported Trap-PDU parameters for SNMPv2C/SNMPv3.

Table 4-2 Supported Trap-PDU parameters (for SNMPv1)

#	Type	Trap-PDU data value					
		enterprise	**agentaddr**	**generic trap**	**specific trap**	**time-stamp**	**variable-bindings**
1	coldStart	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	0	0	sysUpTime value	None
2	warmStart	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	1	0	sysUpTime value	None
3	linkDown	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	2	0	sysUpTime value	ifIndex However, the following MIBs are obtained when `private` is set for a `link_trap_bind_info` parameter by using the `snmp-server -traps` configuration command: ifIndex ifDescr ifType
4	linkUp	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	3	0	sysUpTime value	ifIndex However, the following MIBs are obtained when `private` is set for a `link_trap_bind_info` parameter by using the `snmp-server -traps` configuration command: ifIndex ifDescr ifType

#	Type	Trap-PDU data value					
		enterprise	**agentaddr**	**generic trap**	**specific trap**	**time-stamp**	**variable-bindings**
5	authentication Failure	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	4	0	sysUpTime value	None
6	risingAlarm	Object ID of rmon 1.3.6.1.2.1.16	Specific IP address#	6	1	sysUpTime value	alarmIndex, alarmVariable, alarmSampleType, alarmValue, alarmRisingThreshold
7	fallingAlarm	Object ID of rmon 1.3.6.1.2.1.16	Specific IP address#	6	2	sysUpTime value	alarmIndex, alarmVariable, alarmSampleType, alarmValue, alarmFallingThreshold
8	ax2530sTemperatureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	4	sysUpTime value	ax2530sChassisIndex ax2530sTemperatureStatusIndex ax2530sTemperatureStatusDescr ax2530sTemperatureStatusValue ax2530sTemperatureState
9	ax2530sAxrpStateTransitionTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	40	sysUpTime value	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute axsAxrpGroupMonitoringState
10	ax2530sAxrpMultiFaultDetectionStartTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	41	sysUpTime value	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute
11	ax2530sAxrpMultiFaultDetectionStateTransitionTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	42	sysUpTime value	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute axsAxrpGroupMultiFaultDetectionState

#	Type	Trap-PDU data value					
		enterprise	agentaddr	generic trap	specific trap	time-stamp	variable-bindings
12	ax2530sAirFanStopTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	8	sysUpTime value	None
13	ax2530sPowerSupplyFailureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	9	sysUpTime value	None
14	ax2530sLoginSuccessTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	10	sysUpTime value	axsLoginName, axsLoginTime, axsLoginLocation, axsLoginLine
15	ax2530sLoginFailureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	11	sysUpTime value	axsLoginName, axsLoginFailureTime, axsLoginLocation, axsLoginLine
16	ax2530sLogoutTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	12	sysUpTime value	axsLoginName, axsLoginTime, axsLogoutTime, axsLoginLocation, axsLoginLine, axsLogoutStatus
17	ax2530sBroadcastStormDetectTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	20	sysUpTime value	ifIndex
18	ax2530sMulticastStormDetectTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	21	sysUpTime value	ifIndex
19	ax2530sUnicastStormDetectTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	22	sysUpTime value	ifIndex
20	ax2530sBroadcastStormPortInactivateTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	23	sysUpTime value	ifIndex
21	ax2530sMulticastStormPortInactivateTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	24	sysUpTime value	ifIndex

#	Type	Trap-PDU data value					
		enterprise	agentaddr	generic trap	specific trap	time-stamp	variable-bindings
22	ax2530sUnicastStormPortInactivateTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	25	sysUpTime value	ifIndex
23	ax2530sBroadcastStormRecoverTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	26	sysUpTime value	ifIndex
24	ax2530sMulticastStormRecoverTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	27	sysUpTime value	ifIndex
25	ax2530sUnicastStormRecoverTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	28	sysUpTime value	ifIndex
26	ax2530sEfmoamUdldPortInactivateTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	29	sysUpTime value	ifIndex
27	ax2530sDot1xFailureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	31	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
28	ax2530sDot1xEventTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	32	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
29	ax2530sWauthFailureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	33	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage

#	Type	Trap-PDU data value					
		enterprise	**agentaddr**	**generic trap**	**specific trap**	**time-stamp**	**variable-bindings**
30	ax2530sWauthEventTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	34	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
31	ax2530sMauthFailureTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	35	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
32	ax2530sMauthEventTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	36	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
33	ax2530sDot1xSystemTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	37	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
34	ax2530sWauthSystemTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	38	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
35	ax2530sMauthSystemTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	39	sysUpTime value	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage

#	Type	Trap-PDU data value					
		enterprise	agentaddr	generic trap	specific trap	time-stamp	variable-bindings
36	ax2530sL2ldLinkDown	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	51	sysUpTime value	axsL2ldPortIfIndex axsL2ldPortSourcePortIfindex axsL2ldPortDestinationPortIfindex axsL2ldPortSourceVlan
37	ax2530sL2ldLinkUp	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	52	sysUpTime value	axsL2ldPortIfIndex
38	ax2530sL2ldLoopDetection	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	53	sysUpTime value	axsL2ldPortIndex axsL2ldPortIfIndex axsL2ldPortSourcePortIfindex axsL2ldPortSourceVlan
39	ax2530sUlrChangeSecondary	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	87	sysUpTime value	axsUlrPortIfIndex axsUlrPairedPortIfIndex
40	ax2530sUlrChangePrimary	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	88	sysUpTime value	axsUlrPortIfIndex axsUlrPairedPortIfIndex
41	dot1agCfmFaultAlarm	Object ID of dot1agMIB 1.3.111.2.802.1.1.8	Specific IP address#	6	1	sysUpTime value	dot1agCfmMdIndex dot1agCfmMaIndex dot1agCfmMepIdentifier
42	ax2530sDeviceErrorTrap	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	90	sysUpTime value	ax2530sMemoryError
43	ax2530sSmlStatusFull [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	100	sysUpTime value	ax2530sSmlDomain ax2530sSmlId
44	ax2530sSmlStatusStandalone [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	101	sysUpTime value	ax2530sSmlDomain ax2530sSmlId
45	ax2530sSmlStatusConflict [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	102	sysUpTime value	ax2530sSmlDomain ax2530sSmlId

#	Type	Trap-PDU data value					
		enterprise	agentaddr	generic trap	specific trap	time-stamp	variable-bindings
46	ax2530sSmlPeerlinkNormal [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	103	sysUpTime value	ax2530sSmlDomain ax2530sSmlId
47	ax2530sSmlPeerlinkFailure [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	104	sysUpTime value	ax2530sSmlDomain ax2530sSmlId
48	ax2530sSmlPeerlinkRecovery [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	105	sysUpTime value	ax2530sSmlDomain ax2530sSmlId
49	ax2530sSmlPeerlinkDisconnect [OS-L2A]	sysObjectID of the Switch 1.3.6.1.4.1.21839.1.2.16	Specific IP address#	6	106	sysUpTime value	ax2530sSmlDomain ax2530sSmlId

#

The `agent-addr` value is set in the following order of priority:

1. The IPv4 address set by using the `snmp-server traps agent-address` configuration command.
2. The IPv4 address of an interface in which an IPv4 address is set and that has the lowest ifIndex. The target interface is VLAN.
3. `0.0.0.0` is set if neither of the IPv4 addresses described in 1 and 2 above are set.

Table 4-3 Supported Trap-PDU parameters (for SNMPv2C/SNMPv3)

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	Variable-Binding [2](SnmpTrapOID.0)	Variable-Binding [3-]
1	coldStart	sysUpTime value	Object ID of coldStart (1.3.6.1.6.3.1.1.5.1)	None
2	warmStart	sysUpTime value	Object ID of warmStart (1.3.6.1.6.3.1.1.5.2)	None

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
3	linkDown	sysUpTime value	Object ID of linkDown (1.3.6.1.6.3.1.1.5.3)	ifIndex AdminStatus OperStatus However, the following MIBs are obtained when `private` is set for a `link_trap_bind_info` parameter by using the `snmp-server-traps` configuration command: ifIndex ifDescr ifType
4	linkUp	sysUpTime value	Object ID of linkUp (1.3.6.1.6.3.1.1.5.4)	ifIndex AdminStatus OperStatus However, the following MIBs are obtained when `private` is set for a `link_trap_bind_info` parameter by using the `snmp-server-traps` configuration command: ifIndex ifDescr ifType
5	authentication Failure	sysUpTime value	Object ID of authentication Failure (1.3.6.1.6.3.1.1.5.5)	None
6	risingAlarm	sysUpTime value	Object ID of risingAlarm (1.3.6.1.2.1.16.0.1)	alarmIndex, alarmVariable, alarmSampleType, alarmValue, alarmRisingThreshold
7	fallingAlarm	sysUpTime value	Object ID of fallingAlarm (1.3.6.1.2.1.16.0.2)	alarmIndex, alarmVariable, alarmSampleType, alarmValue, alarmFallingThreshold
8	ax2530sTemperatureTrap	sysUpTime value	Object ID of ax2530sTemperatureTrap (1.3.6.1.4.1.21839.1.2.16.0.4)	ax2530sChassisIndex ax2530sTemperatureStatusIndex ax2530sTemperatureStatusDescr ax2530sTemperatureStatusValue ax2530sTemperatureState
9	ax2530sAxrpStateTransitionTrap	sysUpTime value	Object ID of ax2530sAxrpStateTransitionTrap (1.3.6.1.4.1.21839.1.2.16.0.40)	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute axsAxrpGroupMonitoringState

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
10	ax2530sAxrpMultiFaultDetectionStartTrap	sysUpTime value	Object ID of ax2530sAxrpMultiFaultDetectionStartTrap (1.3.6.1.4.1.21839.1.2.16.0.41)	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute
11	ax2530sAxrpMultiFaultDetectionStateTransitionTrap	sysUpTime value	Object ID of ax2530sAxrpMultiFaultDetectionStateTransitionTrap (1.3.6.1.4.1.21839.1.2.16.0.42)	axsAxrpGroupRingId axsAxrpGroupMode axsAxrpGroupRingAttribute axsAxrpGroupMultiFaultDetectionState
12	ax2530sAirFanStopTrap	sysUpTime value	ax2530sAirFanStopTrap (1.3.6.1.4.1.21839.1.2.16.0.8)	None
13	ax2530sPowerSupplyFailureTrap	sysUpTime value	Object ID of ax2530sPowerSupplyFailureTrap (1.3.6.1.4.1.21839.1.2.16.0.9)	None
14	ax2530sLoginSuccessTrap	sysUpTime value	Object ID of ax2530sLoginSuccessTrap (1.3.6.1.4.1.21839.1.2.16.0.10)	axsLoginName, axsLoginTime, axsLoginLocation, axsLoginLine
15	ax2530sLoginFailureTrap	sysUpTime value	Object ID of ax2530sLoginFailureTrap (1.3.6.1.4.1.21839.1.2.16.0.11)	axsLoginName, axsLoginFailureTime, axsLoginLocation, axsLoginLine
16	ax2530sLogoutTrap	sysUpTime value	Object ID of ax2530sLogoutTrap (1.3.6.1.4.1.21839.1.2.16.0.12)	axsLoginName, axsLoginTime, axsLogoutTime, axsLoginLocation, axsLoginLine, axsLogoutStatus
17	ax2530sBroadcastStormDetectTrap	sysUpTime value	Object ID of ax2530sBsroadcastStormDetectTrap (1.3.6.1.4.1.21839.1.2.16.0.20)	ifIndex
18	ax2530sMulticastStormDetectTrap	sysUpTime value	Object ID of ax2530sMulticastStormDetectTrap (1.3.6.1.4.1.21839.1.2.16.0.21)	ifIndex

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
19	ax2530sUnicastStormDetectTrap	sysUpTime value	Object ID of ax2530sUnicastStormDetectTrap (1.3.6.1.4.1.21839.1.2.16.0.22)	ifIndex
20	ax2530sBroadcastStormPortInactivateTrap	sysUpTime value	Object ID of ax2530sBroadcastStormPortInactivateTrap (1.3.6.1.4.1.21839.1.2.16.0.23)	ifIndex
21	ax2530sMulticastStormPortInactivateTrap	sysUpTime value	Object ID of ax2530sMulticastStormPortInactivateTrap (1.3.6.1.4.1.21839.1.2.16.0.24)	ifIndex
22	ax2530sUnicastStormPortInactivateTrap	sysUpTime value	Object ID of ax2530sUnicastStormPortInactivateTrap (1.3.6.1.4.1.21839.1.2.16.0.25)	ifIndex
23	ax2530sBroadcastStormRecoverTrap	sysUpTime value	Object ID of ax2530sBroadcastStormRecoverTrap (1.3.6.1.4.1.21839.1.2.16.0.26)	ifIndex
24	ax2530sMulticastStormRecoverTrap	sysUpTime value	Object ID of ax2530sMulticastStormRecoverTrap (1.3.6.1.4.1.21839.1.2.16.0.27)	ifIndex
25	ax2530sUnicastStormRecoverTrap	sysUpTime value	Object ID of ax2530sUnicastStormRecoverTrap (1.3.6.1.4.1.21839.1.2.16.0.28)	ifIndex
26	ax2530sEfmoamUdldPortInactivateTrap	sysUpTime value	Object ID of ax2530sEfmoamUdldPortInactivateTrap (1.3.6.1.4.1.21839.1.2.16.0.29)	ifIndex
27	ax2530sDot1xFailureTrap	sysUpTime value	Object ID of ax2530sDot1xFailureTrap (1.3.6.1.4.1.21839.1.2.16.0.31)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
28	ax2530sDot1xEventTrap	sysUpTime value	Object ID of ax2530sDot1xEventTrap (1.3.6.1.4.1.21839.1.2.16.0.32)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
29	ax2530sWauthFailureTrap	sysUpTime value	Object ID of ax2530sWauthFailureTrap (1.3.6.1.4.1.21839.1.2.16.0.33)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
30	ax2530sWauthEventTrap	sysUpTime value	Object ID of ax2530sWauthEventTrap (1.3.6.1.4.1.21839.1.2.16.0.34)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
31	ax2530sMauthFailureTrap	sysUpTime value	Object ID of ax2530sMauthFailureTrap (1.3.6.1.4.1.21839.1.2.16.0.35)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
32	ax2530sMauthEventTrap	sysUpTime value	Object ID of ax2530sMauthEventTrap (1.3.6.1.4.1.21839.1.2.16.0.36)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
33	ax2530sDot1xSystemTrap	sysUpTime value	Object ID of ax2530sDot1xSystemTrap (1.3.6.1.4.1.21839.1.2.16.0.37)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
34	ax2530sWauthSystemTrap	sysUpTime value	Object ID of ax2530sWauthSystemTrap (1.3.6.1.4.1.21839.1.2.16.0.38)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
35	ax2530sMauthSystemTrap	sysUpTime value	Object ID of ax2530sMauthSystemTrap (1.3.6.1.4.1.21839.1.2.16.0.39)	ax2530sAuthSysName ax2530sAuthIfIndex ax2530sAuthSupplicantMac ax2530sAuthMessage
36	ax2530sL2ldLinkDown	sysUpTime value	Object ID of ax2530sL2ldLinkDown (1.3.6.1.4.1.21839.1.2.16.0.51)	axsL2ldPortIfIndex axsL2ldPortSourcePortIfindex axsL2ldPortDestinationPortIfindex axsL2ldPortSourceVlan
37	ax2530sL2ldLinkUp	sysUpTime value	Object ID of ax2530sL2ldLinkUp (1.3.6.1.4.1.21839.1.2.16.0.52)	axsL2ldPortIfIndex

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
38	ax2530sL2ldLoopDetection	sysUpTime value	Object ID of ax2530sMauthEventTrap (1.3.6.1.4.1.21839.1.2.16.0.53)	axsL2ldPortIndex axsL2ldPortIfIndex axsL2ldPortSourcePortIfindex axsL2ldPortSourceVlan
39	ax2530sUlrChangeSecondary	sysUpTime value	Object ID of ax2530sUlrChangeSecondary (1.3.6.1.4.1.21839.1.2.16.0.87)	axsUlrPortIfIndex axsUlrPairedPortIfIndex
40	ax2530sUlrChangePrimary	sysUpTime value	Object ID of ax2530sUlrChangePrimary (1.3.6.1.4.1.21839.1.2.16.0.88)	axsUlrPortIfIndex axsUlrPairedPortIfIndex
41	dot1agCfmFaultAlarm	sysUpTime value	Object ID of dot1agCfmFaultAlarm (1.3.111.2.802.1.1.8.0.1)	dot1agCfmMdIndex dot1agCfmMaIndex dot1agCfmMepIdentifier
42	ax2530sDeviceErrorTrap	sysUpTime value	Object ID of ax2530sDeviceErrorTrap (1.3.6.1.4.1.21839.1.2.16.0.90)	ax2530sMemoryError
43	ax2530sSmlStatusFull [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlStatusFull (1.3.6.1.4.1.21839.1.2.16.0.100)	ax2530sSmlDomain ax2530sSmlId
44	ax2530sSmlStatusStandalone [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlStatusStandalone (1.3.6.1.4.1.21839.1.2.16.0.101)	ax2530sSmlDomain ax2530sSmlId
45	ax2530sSmlStatusConflict [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlStatusConflict (1.3.6.1.4.1.21839.1.2.16.0.102)	ax2530sSmlDomain ax2530sSmlId
46	ax2530sSmlPeerlinkNormal [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlPeerlinkNormal (1.3.6.1.4.1.21839.1.2.16.0.103)	ax2530sSmlDomain ax2530sSmlId
47	ax2530sSmlPeerlinkFailure [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlPeerlinkFailure (1.3.6.1.4.1.21839.1.2.16.0.104)	ax2530sSmlDomain ax2530sSmlId

#	Type	Trap-PDU data value		
		Variable-Binding [1](SysUpTime.0)	**Variable-Binding [2](SnmpTrapOID.0)**	**Variable-Binding [3-]**
48	ax2530sSmlPeerlinkRecovery [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlPeerlinkRecovery (1.3.6.1.4.1.21839.1.2.16.0.105)	ax2530sSmlDomain ax2530sSmlId
49	ax2530sSmlPeerlinkDisconnect [OS-L2A]	sysUpTime value	Object ID of ax2530sSmlPeerlinkDisconnect (1.3.6.1.4.1.21839.1.2.16.0.106)	ax2530sSmlDomain ax2530sSmlId

Appendix

A Private MIB Names and Object ID Values

A. Private MIB Names and Object ID Values

This appendix lists the MIB names and corresponding object ID values of the private MIBs used in the Switch.

A.1 Private MIBs

Private MIB names and their corresponding object ID values are given below.

A.1.1 axsStats group

The MIB names in the axsStats group and their corresponding object ID values are given below.

Table A-1 MIB names in the axsStats group and their corresponding object ID values

MIB name	Object ID
axsStats	1.3.6.1.4.1.21839.2.2.1.1
axsIfStats	1.3.6.1.4.1.21839.2.2.1.1.4
axsIfStatsTable	1.3.6.1.4.1.21839.2.2.1.1.4.1
axsIfStatsEntry	1.3.6.1.4.1.21839.2.2.1.1.4.1.1
axsIfStatsIndex	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.1
axsIfStatsName	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.2
axsIfStatsInMegaOctets	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.3
axsIfStatsInUcastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.4
axsIfStatsInMulticastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.5
axsIfStatsInBroadcastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.6
axsIfStatsOutMegaOctets	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.7
axsIfStatsOutUcastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.8
axsIfStatsOutMulticastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.9
axsIfStatsOutBroadcastMegaPkts	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.10
axsIfStatsHighSpeed	1.3.6.1.4.1.21839.2.2.1.1.4.1.1.11
axsQoS	1.3.6.1.4.1.21839.2.2.1.1.6
axsEtherTxQoS	1.3.6.1.4.1.21839.2.2.1.1.6.1
axsEtherTxQoSStatsTable	1.3.6.1.4.1.21839.2.2.1.1.6.1.1
axsEtherTxQoSStatsEntry	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1
axsEtherTxQoSStatsIndex	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.1

MIB name	Object ID
axsEtherTxQoSStatsMaxQnum	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.2
axsEtherTxQoSStatsLimitQlen	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.3
axsEtherTxQoSStatsTotalOutFrames	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.4
axsEtherTxQoSStatsTotalOutBytesHigh	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.5
axsEtherTxQoSStatsTotalOutBytesLow	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.6
axsEtherTxQoSStatsTotalDiscardFrames	1.3.6.1.4.1.21839.2.2.1.1.6.1.1.1.7
axsEtherTxQoSStatsQueueTable	1.3.6.1.4.1.21839.2.2.1.1.6.1.2
axsEtherTxQoSStatsQueueEntry	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1
axsEtherTxQoSStatsQueueIndex	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.1
axsEtherTxQoSStatsQueueQueIndex	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.2
axsEtherTxQoSStatsQueueQlen	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.3
axsEtherTxQoSStatsQueueMaxQlen	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.4
axsEtherTxQoSStatsQueueDiscardFramesClass1	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.5
axsEtherTxQoSStatsQueueDiscardFramesClass2	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.6
axsEtherTxQoSStatsQueueDiscardFramesClass3	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.7
axsEtherTxQoSStatsQueueDiscardFramesClass4	1.3.6.1.4.1.21839.2.2.1.1.6.1.2.1.8

A.1.2 axsFdb group

The MIB names in the axsFdb group and their corresponding object ID values are given below.

Table A-2 MIB names in the axsFdb group and their corresponding object ID values

MIB name	Object ID
axsFdb	1.3.6.1.4.1.21839.2.2.1.5
axsFdbCounterTable	1.3.6.1.4.1.21839.2.2.1.5.1
axsFdbCounterEntry	1.3.6.1.4.1.21839.2.2.1.5.1.1
axsFdbCounterNifIndex	1.3.6.1.4.1.21839.2.2.1.5.1.1.1
axsFdbCounterLineIndex	1.3.6.1.4.1.21839.2.2.1.5.1.1.2
axsFdbCounterCounts	1.3.6.1.4.1.21839.2.2.1.5.1.1.3

MIB name	Object ID
axsFdbCounterType	1.3.6.1.4.1.21839.2.2.1.5.1.1.4
axsFdbCounterLimits	1.3.6.1.4.1.21839.2.2.1.5.1.1.5

A.1.3 axsVlan group

The MIB names in the axsVlan group and their corresponding object ID values are given below.

Table A-3 MIB names in the axsVlan group and their corresponding object ID values

MIB name	Object ID
axsVlan	1.3.6.1.4.1.21839.2.2.1.6
axsVlanBridge	1.3.6.1.4.1.21839.2.2.1.6.1
axsVlanBridgeBase	1.3.6.1.4.1.21839.2.2.1.6.1.1
axsVBBaseTable	1.3.6.1.4.1.21839.2.2.1.6.1.1.1
axsVBBaseEntry	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1
axsVBBaseIndex	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.1
axsVBBaseBridgeAddress	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.2
axsVBBaseNumPorts	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.3
axsVBBaseType	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.4
axsVBBaseVlanIfIndex	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.5
axsVBBaseVlanType	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.6
axsVBBaseVlanID	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.7
axsVBBaseAssociatedPrimaryVlan	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.8
axsVBBaseIfStatus	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.9
axsVBBaseLastChange	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.10
axsVBBasePrivateVlanType	1.3.6.1.4.1.21839.2.2.1.6.1.1.1.1.11
axsVBBasePortTable	1.3.6.1.4.1.21839.2.2.1.6.1.1.2
axsVBBasePortEntry	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1
axsVBBasePortIndex	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.1
axsVBBasePort	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.2
axsVBBasePortIfIndex	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.3

MIB name	Object ID
axsVBBasePortCircuit	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.4
axsVBBasePortDelayExceededDiscards	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.5
axsVBBasePortMtuExceededDiscards	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.6
axsVBBasePortState	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.7
axsVBBasePortTaggedState	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.8
axsVBBasePortTranslatedTagID	1.3.6.1.4.1.21839.2.2.1.6.1.1.2.1.9
axsVlanBridgeStp	1.3.6.1.4.1.21839.2.2.1.6.1.2
axsVBStpTable	1.3.6.1.4.1.21839.2.2.1.6.1.2.1
axsVBStpEntry	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1
axsVBStpIndex	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.1
axsVBStpProtocolSpecification	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.2
axsVBStpPriority	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.3
axsVBStpTimeSinceTopologyChange	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.4
axsVBStpTopChanges	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.5
axsVBStpDesignatedRoot	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.6
axsVBStpRootCost	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.7
axsVBStpRootPort	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.8
axsVBStpMaxAge	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.9
axsVBStpHelloTime	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.10
axsVBStpHoldTime	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.11
axsVBStpForwardDelay	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.12
axsVBStpBridgeMaxAge	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.13
axsVBStpBridgeHelloTime	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.14
axsVBStpBridgeForwardDelay	1.3.6.1.4.1.21839.2.2.1.6.1.2.1.1.15
axsVBStpPortTable	1.3.6.1.4.1.21839.2.2.1.6.1.2.2
axsVBStpPortEntry	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1
axsVBStpPortIndex	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.1
axsVBStpPort	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.2

MIB name	Object ID
axsVBStpPortPriority	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.3
axsVBStpPortState	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.4
axsVBStpPortEnable	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.5
axsVBStpPortPathCost	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.6
axsVBStpPortDesignatedRoot	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.7
axsVBStpPortDesignatedCost	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.8
axsVBStpPortDesignatedBridge	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.9
axsVBStpPortDesignatedPort	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.10
axsVBStpPortForwardTransitions	1.3.6.1.4.1.21839.2.2.1.6.1.2.2.1.11
axsVlanBridgeTp	1.3.6.1.4.1.21839.2.2.1.6.1.4
axsVBTpTable	1.3.6.1.4.1.21839.2.2.1.6.1.4.1
axsVBTpEntry	1.3.6.1.4.1.21839.2.2.1.6.1.4.1.1
axsVBTpIndex	1.3.6.1.4.1.21839.2.2.1.6.1.4.1.1.1
axsVBTpLearnedEntryDiscards	1.3.6.1.4.1.21839.2.2.1.6.1.4.1.1.2
axsVBTpAgingTime	1.3.6.1.4.1.21839.2.2.1.6.1.4.1.1.3
axsVBTpFdbTable	1.3.6.1.4.1.21839.2.2.1.6.1.4.2
axsVBTpFdbEntry	1.3.6.1.4.1.21839.2.2.1.6.1.4.2.1
axsVBTpFdbIndex	1.3.6.1.4.1.21839.2.2.1.6.1.4.2.1.1
axsVBTpFdbAddress	1.3.6.1.4.1.21839.2.2.1.6.1.4.2.1.2
axsVBTpFdbPort	1.3.6.1.4.1.21839.2.2.1.6.1.4.2.1.3
axsVBTpFdbStatus	1.3.6.1.4.1.21839.2.2.1.6.1.4.2.1.4
axsVBTpPortTable	1.3.6.1.4.1.21839.2.2.1.6.1.4.3
axsVBTpPortEntry	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1
axsVBTpPortIndex	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.1
axsVBTpPort	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.2
axsVBTpPortMaxInfo	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.3
axsVBTpPortInFrames	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.4
axsVBTpPortOutFrames	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.5

MIB name	Object ID
axsVBTpPortInDiscards	1.3.6.1.4.1.21839.2.2.1.6.1.4.3.1.6
axsVlanBridgeStatic	1.3.6.1.4.1.21839.2.2.1.6.1.5
axsVBStaticTable	1.3.6.1.4.1.21839.2.2.1.6.1.5.1
axsVBStaticEntry	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1
axsVBStaticIndex	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1.1
axsVBStaticAddress	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1.2
axsVBStaticReceivePort	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1.3
axsVBStaticAllowedToGoTo	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1.4
axsVBStaticStatus	1.3.6.1.4.1.21839.2.2.1.6.1.5.1.1.5
axsVlanBridgeMaxVlans	1.3.6.1.4.1.21839.2.2.1.6.1.101
axsVlanBridgeMaxSpans	1.3.6.1.4.1.21839.2.2.1.6.1.102
axsVlanTagTranslation	1.3.6.1.4.1.21839.2.2.1.6.10
axsVlanTagTranslationTable	1.3.6.1.4.1.21839.2.2.1.6.10.1
axsVlanTagTranslationEntry	1.3.6.1.4.1.21839.2.2.1.6.10.1.1
axsVlanTagTranslationVlanId	1.3.6.1.4.1.21839.2.2.1.6.10.1.1.1
axsVlanTagTranslationTranslatedId	1.3.6.1.4.1.21839.2.2.1.6.10.1.1.2
axsVlanTagTranslationPorts	1.3.6.1.4.1.21839.2.2.1.6.10.1.1.3

A.1.4 axsL2ldMIB group

The MIB names in the axsL2ldMIB group and their corresponding object ID values are given below.

Table A-4 MIB names in the axsL2ldMIB group and their corresponding object ID values

MIB name	Object ID
axsL2ld	1.3.6.1.4.1.21839.2.2.1.10
axsL2ldGlobalInfo	1.3.6.1.4.1.21839.2.2.1.10.1
axsL2ldVersion	1.3.6.1.4.1.21839.2.2.1.10.1.1
axsL2ldLoopDetectionId	1.3.6.1.4.1.21839.2.2.1.10.1.2
axsL2ldIntervalTime	1.3.6.1.4.1.21839.2.2.1.10.1.3
axsL2ldOutputRate	1.3.6.1.4.1.21839.2.2.1.10.1.4

MIB name	Object ID
axsL2ldThreshold	1.3.6.1.4.1.21839.2.2.1.10.1.5
axsL2ldHoldTime	1.3.6.1.4.1.21839.2.2.1.10.1.6
axsL2ldAutoRestoreTime	1.3.6.1.4.1.21839.2.2.1.10.1.7
axsL2ldConfigurationVlanPortCounts	1.3.6.1.4.1.21839.2.2.1.10.1.8
axsL2ldCapacityVlanPortCounts	1.3.6.1.4.1.21839.2.2.1.10.1.9
axsL2ldPortTable	1.3.6.1.4.1.21839.2.2.1.10.2
axsL2ldPortEntry	1.3.6.1.4.1.21839.2.2.1.10.2.1
axsL2ldPortIndex	1.3.6.1.4.1.21839.2.2.1.10.2.1.1
axsL2ldPortIfIndex	1.3.6.1.4.1.21839.2.2.1.10.2.1.2
axsL2ldPortStatus	1.3.6.1.4.1.21839.2.2.1.10.2.1.3
axsL2ldPortType	1.3.6.1.4.1.21839.2.2.1.10.2.1.4
axsL2ldPortDetectCount	1.3.6.1.4.1.21839.2.2.1.10.2.1.5
axsL2ldPortAutoRestoringTimer	1.3.6.1.4.1.21839.2.2.1.10.2.1.6
axsL2ldPortSourcePortIfindex	1.3.6.1.4.1.21839.2.2.1.10.2.1.7
axsL2ldPortDestinationPortIfindex	1.3.6.1.4.1.21839.2.2.1.10.2.1.8
axsL2ldPortSourceVlan	1.3.6.1.4.1.21839.2.2.1.10.2.1.9
axsL2ldPortHCInFrames	1.3.6.1.4.1.21839.2.2.1.10.2.1.10
axsL2ldPortHCOutFrames	1.3.6.1.4.1.21839.2.2.1.10.2.1.11
axsL2ldPortHCInDiscards	1.3.6.1.4.1.21839.2.2.1.10.2.1.12
axsL2ldPortInactiveCount	1.3.6.1.4.1.21839.2.2.1.10.2.1.13
axsL2ldPortLastInactiveTime	1.3.6.1.4.1.21839.2.2.1.10.2.1.14
axsL2ldPortLastInFramesTime	1.3.6.1.4.1.21839.2.2.1.10.2.1.15

A.1.5 axsUlr group

The MIB names in the axsUlr group and their corresponding object ID values are given below.

Table A-5 MIB names in the axsUlr group and their corresponding object ID values

MIB name	Object ID
axsUlr	1.3.6.1.4.1.21839.2.2.1.20

MIB name	Object ID
axsUlrGlobalInfo	1.3.6.1.4.1.21839.2.2.1.20.1
axsUlrVersion	1.3.6.1.4.1.21839.2.2.1.20.1.1
axsUlrID	1.3.6.1.4.1.21839.2.2.1.20.1.2
axsUlrConfigurationPortCounts	1.3.6.1.4.1.21839.2.2.1.20.1.3
axsUlrStartupActivePortSelection	1.3.6.1.4.1.21839.2.2.1.20.1.4
axsUlrPortTable	1.3.6.1.4.1.21839.2.2.1.20.2
axsUlrPortEntry	1.3.6.1.4.1.21839.2.2.1.20.2.1
axsUlrPortIfIndex	1.3.6.1.4.1.21839.2.2.1.20.2.1.1
axsUlrPortType	1.3.6.1.4.1.21839.2.2.1.20.2.1.2
axsUlrPairedPortIfIndex	1.3.6.1.4.1.21839.2.2.1.20.2.1.3
axsUlrPortStatus	1.3.6.1.4.1.21839.2.2.1.20.2.1.4
axsUlrPairedPortStatus	1.3.6.1.4.1.21839.2.2.1.20.2.1.5
axsUlrAutoChangeToPrimary	1.3.6.1.4.1.21839.2.2.1.20.2.1.6
axsUlrAutoChangeToPrimaryDelay	1.3.6.1.4.1.21839.2.2.1.20.2.1.7
axsUlrAutoChangeToPrimaryRest	1.3.6.1.4.1.21839.2.2.1.20.2.1.8
axsUlrStartupActivePortSelectionStatus	1.3.6.1.4.1.21839.2.2.1.20.2.1.9
axsUlrFlushTransmit	1.3.6.1.4.1.21839.2.2.1.20.2.1.10
axsUlrFlushVlan	1.3.6.1.4.1.21839.2.2.1.20.2.1.11
axsUlrMacAddressUpdateTransmit	1.3.6.1.4.1.21839.2.2.1.20.2.1.12
axsUlrLastActivePortDecisionTime	1.3.6.1.4.1.21839.2.2.1.20.2.1.13
axsUlrLastFlushTransmitTime	1.3.6.1.4.1.21839.2.2.1.20.2.1.14
axsUlrLastMacUpdateTransmitTime	1.3.6.1.4.1.21839.2.2.1.20.2.1.15
axsUlrLastChangeFactor	1.3.6.1.4.1.21839.2.2.1.20.2.1.16
axsUlrFlushTransmitTotalPackets	1.3.6.1.4.1.21839.2.2.1.20.2.1.17
axsUlrMacAddressUpdateTransmitTotalPackets	1.3.6.1.4.1.21839.2.2.1.20.2.1.18
axsUlrMacAddressUpdateTransmitOverFlow	1.3.6.1.4.1.21839.2.2.1.20.2.1.19
axsUlrActiveDecisionCount	1.3.6.1.4.1.21839.2.2.1.20.2.1.20

A.1.6 axsBootManagement group

The MIB names in the axsBootManagement group and their corresponding object ID values are given below.

Table A-6 MIB names in the axsBootManagement group and their corresponding object ID values

MIB name	Object ID
axsBootManagement	1.3.6.1.4.1.21839.2.2.1.51
axsBootReason	1.3.6.1.4.1.21839.2.2.1.51.1

A.1.7 axsLogin group

The MIB names in the axsLogin group and their corresponding object ID values are given below.

Table A-7 MIB names in the axsLogin group and their corresponding object ID values

MIB name	Object ID
axsLogin	1.3.6.1.4.1.21839.2.2.1.52
axsLoginName	1.3.6.1.4.1.21839.2.2.1.52.1
axsLoginTime	1.3.6.1.4.1.21839.2.2.1.52.2
axsLogoutTime	1.3.6.1.4.1.21839.2.2.1.52.3
axsLoginFailureTime	1.3.6.1.4.1.21839.2.2.1.52.4
axsLoginLocation	1.3.6.1.4.1.21839.2.2.1.52.5
axsLoginLine	1.3.6.1.4.1.21839.2.2.1.52.6
axsLogoutStatus	1.3.6.1.4.1.21839.2.2.1.52.7

A.1.8 axslldp group

The MIB names in the axslldp group and their corresponding object ID values are given below.

Table A-8 MIB names in the axslldp group and their corresponding object ID values

MIB name	Object ID
axslldp	1.3.6.1.4.1.21839.2.2.1.100
axslldpConfiguration	1.3.6.1.4.1.21839.2.2.1.100.1
axslldpMessageTxInterval	1.3.6.1.4.1.21839.2.2.1.100.1.1
axslldpMessageTxHoldMultiplier	1.3.6.1.4.1.21839.2.2.1.100.1.2
axslldpReinitDelay	1.3.6.1.4.1.21839.2.2.1.100.1.3

MIB name	Object ID
axslldpTxDelay	1.3.6.1.4.1.21839.2.2.1.100.1.4
axslldpPortConfigTable	1.3.6.1.4.1.21839.2.2.1.100.1.6
axslldpPortConfigEntry	1.3.6.1.4.1.21839.2.2.1.100.1.6.1
axslldpPortConfigPortNum	1.3.6.1.4.1.21839.2.2.1.100.1.6.1.2
axslldpPortConfigAdminStatus	1.3.6.1.4.1.21839.2.2.1.100.1.6.1.3
axslldpPortConfigTLVsTxEnable	1.3.6.1.4.1.21839.2.2.1.100.1.6.1.4
axslldpPortConfigRowStatus	1.3.6.1.4.1.21839.2.2.1.100.1.6.1.5
axslldpConfigManAddrTable	1.3.6.1.4.1.21839.2.2.1.100.1.7
axslldpConfigManAddrEntry	1.3.6.1.4.1.21839.2.2.1.100.1.7.1
axslldpConfigManAddrPortsTxEnable	1.3.6.1.4.1.21839.2.2.1.100.1.7.1.1
axslldpStats	1.3.6.1.4.1.21839.2.2.1.100.2
axslldpStatsTable	1.3.6.1.4.1.21839.2.2.1.100.2.1
axslldpStatsEntry	1.3.6.1.4.1.21839.2.2.1.100.2.1.1
axslldpStatsPortNum	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.2
axslldpStatsOperStatus	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.3
axslldpStatsFramesInErrors	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.4
axslldpStatsFramesInTotal	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.5
axslldpStatsFramesOutTotal	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.6
axslldpStatsTLVsInErrors	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.7
axslldpStatsTLVsDiscardedTotal	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.8
axslldpStatsCounterDiscontinuityTime	1.3.6.1.4.1.21839.2.2.1.100.2.1.1.9
axslldpLocalSystemData	1.3.6.1.4.1.21839.2.2.1.100.3
axslldpLocChassisType	1.3.6.1.4.1.21839.2.2.1.100.3.1
axslldpLocChassisId	1.3.6.1.4.1.21839.2.2.1.100.3.2
axslldpLocSysName	1.3.6.1.4.1.21839.2.2.1.100.3.3
axslldpLocSysDesc	1.3.6.1.4.1.21839.2.2.1.100.3.4
axslldpLocSysCapSupported	1.3.6.1.4.1.21839.2.2.1.100.3.5
axslldpLocSysCapEnabled	1.3.6.1.4.1.21839.2.2.1.100.3.6

MIB name	Object ID
axslldpLocPortTable	1.3.6.1.4.1.21839.2.2.1.100.3.7
axslldpLocPortEntry	1.3.6.1.4.1.21839.2.2.1.100.3.7.1
axslldpLocPortNum	1.3.6.1.4.1.21839.2.2.1.100.3.7.1.1
axslldpLocPortType	1.3.6.1.4.1.21839.2.2.1.100.3.7.1.2
axslldpLocPortId	1.3.6.1.4.1.21839.2.2.1.100.3.7.1.3
axslldpLocPortDesc	1.3.6.1.4.1.21839.2.2.1.100.3.7.1.4
axslldpLocManAddrTable	1.3.6.1.4.1.21839.2.2.1.100.3.8
axslldpLocManAddrEntry	1.3.6.1.4.1.21839.2.2.1.100.3.8.1
axslldpLocManAddrType	1.3.6.1.4.1.21839.2.2.1.100.3.8.1.1
axslldpLocManAddr	1.3.6.1.4.1.21839.2.2.1.100.3.8.1.1.2
axslldpLocManAddrIfSubtype	1.3.6.1.4.1.21839.2.2.1.100.3.8.1.1.3
axslldpLocManAddrIfId	1.3.6.1.4.1.21839.2.2.1.100.3.8.1.1.4
axslldpLocManAddrOID	1.3.6.1.4.1.21839.2.2.1.100.3.8.1.1.5
axslldpRemoteSystemData	1.3.6.1.4.1.21839.2.2.1.100.4
axslldpRemTable	1.3.6.1.4.1.21839.2.2.1.100.4.1
axslldpRemEntry	1.3.6.1.4.1.21839.2.2.1.100.4.1.1
axslldpRemTimeMark	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.1
axslldpRemLocalPortNum	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.2
axslldpRemIndex	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.3
axslldpRemRemoteChassisType	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.4
axslldpRemRemoteChassis	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.5
axslldpRemRemotePortType	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.6
axslldpRemRemotePort	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.7
axslldpRemPortDesc	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.8
axslldpRemSysName	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.9
axslldpRemSysDesc	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.10
axslldpRemSysCapSupported	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.11
axslldpRemSysCapEnabled	1.3.6.1.4.1.21839.2.2.1.100.4.1.1.12

MIB name	Object ID
axslldpRemManAddrTable	1.3.6.1.4.1.21839.2.2.1.100.4.2
axslldpRemManAddrEntry	1.3.6.1.4.1.21839.2.2.1.100.4.2.1
axslldpRemManAddrType	1.3.6.1.4.1.21839.2.2.1.100.4.2.1.1
axslldpRemManAddr	1.3.6.1.4.1.21839.2.2.1.100.4.2.1.2
axslldpRemManAddrIfSubtype	1.3.6.1.4.1.21839.2.2.1.100.4.2.1.3
axslldpRemManAddrIfId	1.3.6.1.4.1.21839.2.2.1.100.4.2.1.4
axslldpRemManAddrOID	1.3.6.1.4.1.21839.2.2.1.100.4.2.1.5
axslldpRemOrgDefInfoTable	1.3.6.1.4.1.21839.2.2.1.100.4.3
axslldpRemOrgDefInfoEntry	1.3.6.1.4.1.21839.2.2.1.100.4.3.1
axslldpRemOrgDefInfoOUI	1.3.6.1.4.1.21839.2.2.1.100.4.3.1.1
axslldpRemOrgDefInfoSubtype	1.3.6.1.4.1.21839.2.2.1.100.4.3.1.2
axslldpRemOrgDefInfoIndex	1.3.6.1.4.1.21839.2.2.1.100.4.3.1.3
axslldpRemOrgDefInfo	1.3.6.1.4.1.21839.2.2.1.100.4.3.1.4
axslldpRemoteOriginInfoData	1.3.6.1.4.1.21839.2.2.1.100.20
axslldpRemOriginInfoTable	1.3.6.1.4.1.21839.2.2.1.100.20.1
axslldpRemOriginInfoEntry	1.3.6.1.4.1.21839.2.2.1.100.20.1.1
axslldpRemOriginInfoPortNum	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.1
axslldpRemOriginInfoIndex	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.2
axslldpRemOriginInfoLowerVlanList	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.3
axslldpRemOriginInfoHigherVlanList	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.4
axslldpRemOriginInfoIPv4Address	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.5
axslldpRemOriginInfoIPv4PortType	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.6
axslldpRemOriginInfoIPv4VlanId	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.7
axslldpRemOriginInfoIPv6Address	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.8
axslldpRemOriginInfoIPv6PortType	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.9
axslldpRemOriginInfoIPv6VlanId	1.3.6.1.4.1.21839.2.2.1.100.20.1.1.10

A.1.9 axsAxrpMIB group

The MIB names in the axsAxrpMIB group and their corresponding object ID values are given below.

Table A-9 MIB names in the axsAxrpMIB group and their corresponding object ID values

MIB name	Object ID
axsAxrp	1.3.6.1.4.1.21839.2.2.1.200
axsAxrpGroupTable	1.3.6.1.4.1.21839.2.2.1.200.1
axsAxrpGroupEntry	1.3.6.1.4.1.21839.2.2.1.200.1.1
axsAxrpGroupRingId	1.3.6.1.4.1.21839.2.2.1.200.1.1.1
axsAxrpGroupRowStatus	1.3.6.1.4.1.21839.2.2.1.200.1.1.2
axsAxrpGroupMode	1.3.6.1.4.1.21839.2.2.1.200.1.1.3
axsAxrpGroupRingAttribute	1.3.6.1.4.1.21839.2.2.1.200.1.1.4
axsAxrpGroupMonitoringState	1.3.6.1.4.1.21839.2.2.1.200.1.1.5
axsAxrpGroupRingport1	1.3.6.1.4.1.21839.2.2.1.200.1.1.6
axsAxrpGroupRingport1Shared	1.3.6.1.4.1.21839.2.2.1.200.1.1.7
axsAxrpGroupRingport2	1.3.6.1.4.1.21839.2.2.1.200.1.1.8
axsAxrpGroupRingport2Shared	1.3.6.1.4.1.21839.2.2.1.200.1.1.9
axsAxrpGroupTransitionToFaultCounts	1.3.6.1.4.1.21839.2.2.1.200.1.1.10
axsAxrpGroupTransitionToNormalCounts	1.3.6.1.4.1.21839.2.2.1.200.1.1.11
axsAxrpGroupLastTransitionTime	1.3.6.1.4.1.21839.2.2.1.200.1.1.12
axsAxrpGroupMultiFaultDetectionState	1.3.6.1.4.1.21839.2.2.1.200.1.1.22
axsAxrpVlanGroupTable	1.3.6.1.4.1.21839.2.2.1.200.2
axsAxrpVlanGroupEntry	1.3.6.1.4.1.21839.2.2.1.200.2.1
axsAxrpVlanGroupRingId	1.3.6.1.4.1.21839.2.2.1.200.2.1.1
axsAxrpVlanGroupId	1.3.6.1.4.1.21839.2.2.1.200.2.1.2
axsAxrpVlanGroupRingport1	1.3.6.1.4.1.21839.2.2.1.200.2.1.3
axsAxrpVlanGroupRingport1Role	1.3.6.1.4.1.21839.2.2.1.200.2.1.4
axsAxrpVlanGroupRingport1OperState	1.3.6.1.4.1.21839.2.2.1.200.2.1.5
axsAxrpVlanGroupRingport2	1.3.6.1.4.1.21839.2.2.1.200.2.1.6
axsAxrpVlanGroupRingport2Role	1.3.6.1.4.1.21839.2.2.1.200.2.1.7

MIB name	Object ID
axsAxrpVlanGroupRingport2OperState	1.3.6.1.4.1.21839.2.2.1.200.2.1.8

A.1.10 axsPconMIB group

The MIB names in the axsPconMIB group and their corresponding object ID values are given below.

Table A-10 MIB names in the axsPconMIB group and their corresponding object ID values

MIB name	Object ID
axsPconMIB	1.3.6.1.4.1.21839.2.2.1.300
axsPconObjects	1.3.6.1.4.1.21839.2.2.1.300.1
axsPconModuleData	1.3.6.1.4.1.21839.2.2.1.300.1.1
axsPconModuleTable	1.3.6.1.4.1.21839.2.2.1.300.1.1.1
axsPconModuleEntry	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1
axsPconModuleIndex	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.1
axsPconModuleType	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.2
axsPconModuleSlotNo	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.3
axsPconModuleDescr	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.4
axsPconModuleStatus	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.5
axsPconModuleMode	1.3.6.1.4.1.21839.2.2.1.300.1.1.1.1.6
axsPconPowerCon	1.3.6.1.4.1.21839.2.2.1.300.1.2
axsPconPowerConTable	1.3.6.1.4.1.21839.2.2.1.300.1.2.1
axsPconPowerConEntry	1.3.6.1.4.1.21839.2.2.1.300.1.2.1.1
axsPconPowerConMaxPower	1.3.6.1.4.1.21839.2.2.1.300.1.2.1.1.1
axsPconPowerConPowerConsumption	1.3.6.1.4.1.21839.2.2.1.300.1.2.1.1.2
axsPconPowerConPowerMeter	1.3.6.1.4.1.21839.2.2.1.300.1.2.1.1.3
axsPconTraffic	1.3.6.1.4.1.21839.2.2.1.300.1.3
axsPconTrafficTable	1.3.6.1.4.1.21839.2.2.1.300.1.3.1
axsPconTrafficEntry	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1
axsPconTrafficMaxTransferCapacity	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.1
axsPconTrafficTotalTransferCapacity	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.2

MIB name	Object ID
axsPconTrafficInOctets	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.3
axsPconTrafficOutOctets	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.4
axsPconTrafficInPkts	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.5
axsPconTrafficOutPkts	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.6
axsPconTrafficCapacityOctets	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.7
axsPconTrafficInPeakOctetsRate	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.8
axsPconTrafficPeakTransferCapacity	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.9
axsPconTrafficInDiscPkts	1.3.6.1.4.1.21839.2.2.1.300.1.3.1.1.10

A.1.11 ax2530sSwitch group

The MIB names in the ax2530sSwitch group and their corresponding object ID values are given below.

Table A-11 MIB names in the ax2530sSwitch group and their corresponding object ID values

MIB name	Object ID
ax2530sSwitch	1.3.6.1.4.1.21839.2.2.16.1
ax2530sModelType	1.3.6.1.4.1.21839.2.2.16.1.1
ax2530sSoftware	1.3.6.1.4.1.21839.2.2.16.1.2
ax2530sSoftwareName	1.3.6.1.4.1.21839.2.2.16.1.2.1
ax2530sSoftwareAbbreviation	1.3.6.1.4.1.21839.2.2.16.1.2.2
ax2530sSoftwareVersion	1.3.6.1.4.1.21839.2.2.16.1.2.3
ax2530sSystemMsg	1.3.6.1.4.1.21839.2.2.16.1.3
ax2530sSystemMsgText	1.3.6.1.4.1.21839.2.2.16.1.3.1
ax2530sSystemMsgType	1.3.6.1.4.1.21839.2.2.16.1.3.2
ax2530sSystemMsgTimeStamp	1.3.6.1.4.1.21839.2.2.16.1.3.3
ax2530sSystemMsgLevel	1.3.6.1.4.1.21839.2.2.16.1.3.4
ax2530sSystemMsgEventPoint	1.3.6.1.4.1.21839.2.2.16.1.3.5
ax2530sSystemMsgEventInterfaceID	1.3.6.1.4.1.21839.2.2.16.1.3.6
ax2530sSystemMsgEventCode	1.3.6.1.4.1.21839.2.2.16.1.3.7
ax2530sSystemMsgAdditionalCode	1.3.6.1.4.1.21839.2.2.16.1.3.8

MIB name	Object ID
ax2530sSnmpAgent	1.3.6.1.4.1.21839.2.2.16.1.4
ax2530sSnmpSendReceiveSize	1.3.6.1.4.1.21839.2.2.16.1.4.1
ax2530sSnmpReceiveDelay	1.3.6.1.4.1.21839.2.2.16.1.4.2
ax2530sSnmpContinuousSend	1.3.6.1.4.1.21839.2.2.16.1.4.3
ax2530sSnmpObjectMaxNumber	1.3.6.1.4.1.21839.2.2.16.1.4.4
ax2530sLicense	1.3.6.1.4.1.21839.2.2.16.1.6
ax2530sLicenseNumber	1.3.6.1.4.1.21839.2.2.16.1.6.1
ax2530sLicenseTable	1.3.6.1.4.1.21839.2.2.16.1.6.2
ax2530sLicenseEntry	1.3.6.1.4.1.21839.2.2.16.1.6.2.1
ax2530sLicenseIndex	1.3.6.1.4.1.21839.2.2.16.1.6.2.1.1
ax2530sLicenseSerialNumber	1.3.6.1.4.1.21839.2.2.16.1.6.2.1.2
ax2530sLicenseOptionNumber	1.3.6.1.4.1.21839.2.2.16.1.6.2.1.3
ax2530sLicenseOptionTable	1.3.6.1.4.1.21839.2.2.16.1.6.3
ax2530sLicenseOptionEntry	1.3.6.1.4.1.21839.2.2.16.1.6.3.1
ax2530sLicenseOptionIndex	1.3.6.1.4.1.21839.2.2.16.1.6.3.1.1
ax2530sLicenseOptionNumberIndex	1.3.6.1.4.1.21839.2.2.16.1.6.3.1.2
ax2530sLicenseOptionSoftwareName	1.3.6.1.4.1.21839.2.2.16.1.6.3.1.3
ax2530sLicenseOptionSoftwareAbbreviation	1.3.6.1.4.1.21839.2.2.16.1.6.3.1.4

A.1.12 ax2530sDevice group

The MIB names in the ax2530sDevice group and their corresponding object ID values are given below.

Table A-12 MIB names in the ax2530sDevice group and their corresponding object ID values

MIB name	Object ID
ax2530sDevice	1.3.6.1.4.1.21839.2.2.16.2
ax2530sChassis	1.3.6.1.4.1.21839.2.2.16.2.1
ax2530sChassisMaxNumber	1.3.6.1.4.1.21839.2.2.16.2.1.1
ax2530sChassisTable	1.3.6.1.4.1.21839.2.2.16.2.1.2
ax2530sChassisEntry	1.3.6.1.4.1.21839.2.2.16.2.1.2.1

MIB name	Object ID
ax2530sChassisIndex	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.1
ax2530sChassisType	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.2
ax2530sChassisStatus	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.3
ax2530sStsLedStatus	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.4
ax2530sMemoryTotalSize	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.7
ax2530sMemoryUsedSize	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.8
ax2530sMemoryFreeSize	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.9
ax2530sCpuLoad1m	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.11
ax2530sPhysLineNumber	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.19
ax2530sTemperatureStatusNumber	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.20
ax2530sPowerUnitNumber	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.21
ax2530sRedundantPsNumber	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.22
ax2530sFanNumber	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.23
ax2530sTotalAccumRunTime	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.24
ax2530sCriticalAccumRunTime	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.25
ax2530sSts2LedStatus	1.3.6.1.4.1.21839.2.2.16.2.1.2.1.26
ax2530sTemperatureStatusTable	1.3.6.1.4.1.21839.2.2.16.2.1.3
ax2530sTemperatureStatusEntry	1.3.6.1.4.1.21839.2.2.16.2.1.3.1
ax2530sTemperatureStatusIndex	1.3.6.1.4.1.21839.2.2.16.2.1.3.1.1
ax2530sTemperatureStatusDescr	1.3.6.1.4.1.21839.2.2.16.2.1.3.1.2
ax2530sTemperatureStatusValue	1.3.6.1.4.1.21839.2.2.16.2.1.3.1.3
ax2530sTemperatureThreshold	1.3.6.1.4.1.21839.2.2.16.2.1.3.1.4
ax2530sTemperatureState	1.3.6.1.4.1.21839.2.2.16.2.1.3.1.5
ax2530sPowerUnitTable	1.3.6.1.4.1.21839.2.2.16.2.1.4
ax2530sPowerUnitEntry	1.3.6.1.4.1.21839.2.2.16.2.1.4.1
ax2530sPowerUnitIndex	1.3.6.1.4.1.21839.2.2.16.2.1.4.1.1
ax2530sPowerConnectStatus	1.3.6.1.4.1.21839.2.2.16.2.1.4.1.2
ax2530sPowerSupplyStatus	1.3.6.1.4.1.21839.2.2.16.2.1.4.1.3

MIB name	Object ID
ax2530sRedundantPsTable	1.3.6.1.4.1.21839.2.2.16.2.1.6
ax2530sFanTable	1.3.6.1.4.1.21839.2.2.16.2.1.5
ax2530sFanEntry	1.3.6.1.4.1.21839.2.2.16.2.1.5.1
ax2530sFanIndex	1.3.6.1.4.1.21839.2.2.16.2.1.5.1.1
ax2530sFanStatus	1.3.6.1.4.1.21839.2.2.16.2.1.5.1.2
ax2530sRedundantPsEntry	1.3.6.1.4.1.21839.2.2.16. 2.1.6.1
ax2530sRedundantPsIndex	1.3.6.1.4.1.21839.2.2.16.2.1.6.1.1
ax2530sRedundantPsConnectStatus	1.3.6.1.4.1.21839.2.2.16.2.1.6.1.2
ax2530sRedundantPsStatus	1.3.6.1.4.1.21839.2.2.16.2.1.6.1.3
ax2530sPhysLine	1.3.6.1.4.1.21839.2.2.16.2.2
ax2530sPhysLineTable	1.3.6.1.4.1.21839.2.2.16.2.2.1
ax2530sPhysLineEntry	1.3.6.1.4.1.21839.2.2.16.2.2.1.1
ax2530sPhysLineIndex	1.3.6.1.4.1.21839.2.2.16.2.2.1.1.1
ax2530sPhysLineConnectorType	1.3.6.1.4.1.21839.2.2.16.2.2.1.1.2
ax2530sPhysLineOperStatus	1.3.6.1.4.1.21839.2.2.16.2.2.1.1.3
ax2530sPhysLineIfIndexNumber	1.3.6.1.4.1.21839.2.2.16.2.2.1.1.4
ax2530sPhysLineTransceiverStatus	1.3.6.1.4.1.21839.2.2.16.2.2.1.1.5
ax2530sDeviceError	1.3.6.1.4.1.21839.2.2.16.2.3
ax2530sMemoryError	1.3.6.1.4.1.21839.2.2.16.2.3.1

A.1.13 ax2530sAuth group

The MIB names in the ax2530sAuth group and their corresponding object ID values are given below.

Table A-13 MIB names in the ax2530sAuth group and their corresponding object ID values

MIB name	Object ID
ax2530sAuth	1.3.6.1.4.1.21839.2.2.16.10
ax2530sAuthInfo	1.3.6.1.4.1.21839.2.2.16.10.1
ax2530sAuthSysName	1.3.6.1.4.1.21839.2.2.16.10.1.1
ax2530sAuthIfIndex	1.3.6.1.4.1.21839.2.2.16.10.1.2

MIB name	Object ID
ax2530sAuthSupplicantMac	1.3.6.1.4.1.21839.2.2.16.10.1.3
ax2530sAuthMessage	1.3.6.1.4.1.21839.2.2.16.10.1.4

A.1.14 ax2530sSml group [OS-L2A]

The MIB names in the ax2530sSml group and their corresponding object ID values are given below.

Table A-14 MIB names in the ax2530sSml group and their corresponding object ID values

MIB name	Object ID
ax2530sSml	1.3.6.1.4.1.21839.2.2.16.20
ax2530sSmlInfo	1.3.6.1.4.1.21839.2.2.16.20.1
ax2530sSmlDomain	1.3.6.1.4.1.21839.2.2.16.20.1.1
ax2530sSmlId	1.3.6.1.4.1.21839.2.2.16.20.1.2

A.2 InMon private MIBs

InMon private MIB names and their corresponding object ID values are given below.

Table A-15 InMon private MIB names and their corresponding object ID values

MIB name	Object ID
sFlowMIB	1.3.6.1.4.1.4300.1
sFlowAgent	1.3.6.1.4.1.4300.1.1
sFlowVersion	1.3.6.1.4.1.4300.1.1.1
sFlowAgentAddressType	1.3.6.1.4.1.4300.1.1.2
sFlowAgentAddress	1.3.6.1.4.1.4300.1.1.3
sFlowTable	1.3.6.1.4.1.4300.1.1.4
sFlowEntry	1.3.6.1.4.1.4300.1.1.4.1
sFlowDataSource	1.3.6.1.4.1.4300.1.1.4.1.1
sFlowOwner	1.3.6.1.4.1.4300.1.1.4.1.2
sFlowTimeout	1.3.6.1.4.1.4300.1.1.4.1.3
sFlowPacketSamplingRate	1.3.6.1.4.1.4300.1.1.4.1.4
sFlowCounterSamplingInterval	1.3.6.1.4.1.4300.1.1.4.1.5
sFlowMaximumHeaderSize	1.3.6.1.4.1.4300.1.1.4.1.6

MIB name	Object ID
sFlowMaximumDatagramSize	1.3.6.1.4.1.4300.1.1.4.1.7
sFlowCollectorAddressType	1.3.6.1.4.1.4300.1.1.4.1.8
sFlowCollectorAddress	1.3.6.1.4.1.4300.1.1.4.1.9
sFlowCollectorPort	1.3.6.1.4.1.4300.1.1.4.1.10
sFlowDatagramVersion	1.3.6.1.4.1.4300.1.1.4.1.11

Index