



## **Remote Management of MoCA Interfaces using SNMP MIB**

### **Reference document**

**Approved DRAFT DELIVERABLE MoCA Access-MIB-V2.5-20180523**



## Table of Contents

1	Introduction.....	4
2	Data Model .....	4
3	Object Definitions.....	5
3.1	Object Template.....	6
3.1.1	Interface Configuration Table .....	7
3.1.2	Interface RLAPM Table.....	11
3.1.3	Interface SAPM Table.....	12
3.1.4	Interface ACA Table .....	12
3.1.5	Interface Status Table.....	13
3.1.6	Interface Statistics Table .....	17
3.1.7	Link Statistics Table.....	18
3.2	Network Related Objects .....	18
3.2.1	Node Table.....	19
3.2.2	P2mp Table .....	20
3.2.3	Link Modulation Table.....	20
3.3	Notification Events .....	21
4	Appendix A.....	23
4.1	Aca Power Profile .....	23
4.1.1	Encoding for the Object mocaIfAcaPowerProfile .....	23
4.1.2	Encoding for the Data Type MocaScModList .....	25
4.2	Authentication Exempt.....	26
4.3	Authentication Result.....	26
4.4	Link Parameters .....	27
4.5	Net Parameters .....	27
4.6	Node Parameters .....	27
4.7	Connected Node Info .....	28
4.8	Flow Info.....	28
4.9	PerCent Priority .....	28
4.10	P2mpPhyRates .....	29

4.11 Software Version..... 29

4.12 Link State..... 31

4.13 Node Power..... 31

4.14 Node RX SNR..... 31

4.15 Node Power Reduction ..... 32

5 Appendix B..... 32

5.1 Algorithm to calculate MoCA Password Hash..... 32

5.2 Example ..... 33

6 MoCA Access 2.5-MIB Definition ..... 34

7 Acknowledgements..... 103

8 References..... 103

### List of Figures

Figure 1 Relationship Diagram between MIB Module Tables..... 5

### List of Tables

Table 1: mocaIfConfigTable..... 7

Table 2: mocaIfRlapmTable ..... 11

Table 3: mocaIfSapmTable..... 12

Table 4: mocaIfAcaTable ..... 12

Table 5: mocaIfStatusTable ..... 13

Table 6: mocaIfStatsTable ..... 17

Table 7: mocaLinkStatsTable ..... 18

Table 8: mocaNodeTable ..... 19

Table 9: mocaP2mpTable ..... 20

Table 10: mocaScModTable ..... 20

Table 11: mocaNotifications..... 21

Table 12: MoCA Password Hash Example ..... 33

## 1 Introduction

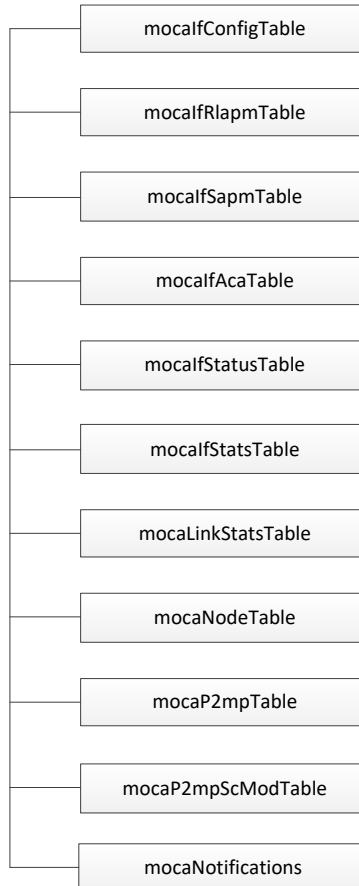
This document defines a Multimedia over Coax Alliance Management Information Base (MoCA\_ACCESS\_25-MIB) to remotely configure, monitor, and diagnose MOCA2.5 Access devices which supports one or more MoCA interfaces. The MoCA\_ACCESS\_25-MIB must be used with the companion memo MoCA Enterprise Structure of Management Information (MoCA-SMI) [4].

## 2 Data Model

The MoCA Access 2.5 data model conforms to the Structure of Management Information Version 2 (SMIv2) [1][2][3]. The model is a collection of managed objects termed a Management Information Base (MIB). The MoCA MIB is comprised of tabular objects (i.e., tables) for status reporting and configuration of the managed MoCA interfaces. In addition, Notifications are also defined to allow dynamic recording and/or reporting exceptional conditions or events that may warrant immediate action by the network Management System operators. The MIB objects defined in this document are intended to be used by MoCA interfaces that support MoCA Access Specification v2.5 [5].

### 3 Object Definitions

MoCA-MIB objects are organized into 11 tables in the MIB module. Each table is at least indexed by the interface index (ifIndex).



**Figure 1 Relationship Diagram between MIB Module Tables**

Since a MoCA device can have one or more MoCA interfaces, each MoCA interface is uniquely identified by the interface index (ifIndex). The MIB module also supports MoCA interface initiated notification events.

The first ten tables are organized into two main categories,

1. MIB objects specific to the managed MoCA Node only.
2. MIB objects related to the MoCA network which the managed MoCA Node is part of.

The final table contains the notification events.

This section uses object templates to provide an overview of the managed objects.

### 3.1 Object Template

There is one object template to tabulate each MIB table. There is an additional object template to tabulate all the notification events. Each object template has the same seven columns, which are:

1. ID. This is the last digit before the instance indices in the MIB Object Identifier (OID).
2. Object Name. This is the name of the object as appears in the MIB module.
3. Type. The data types are defined in Table A1.

**Table A1: Data Types**

Data Type	Description
INTEGER	Integer with enumerated values.
Integer32	32 bit signed integer (−2,147,483,648 to +2,147,483,647)
Unsigned32	32 bit unsigned integer (0 to 4,294,967,295)
Boolean	The allowed values are “0”, “1”, “true”, and “false”. The values “1” and “true” are considered interchangeable, where both equivalently represent the logical value <i>true</i> . Similarly, the values “0” and “false” are considered interchangeable, where both equivalently represent the logical value <i>false</i> .
BITS	String of named bits.
OCTET STRING	String of octets (eight-bit values) with no restrictions on the value of any octet.
MacAddress	An OCTET STRING of size six which represents an IEEE 802 MAC address.
Counter32	32 bit unsigned integer (0 to 4,294,967,295)
Counter64	64 bit unsigned integer (0 to 18,446,744,073,709,551,615)
DisplayString	String with only ASCII characters for display purposes.
RowStatus	Integer used to manage the creation and deletion of conceptual rows.

4. Req't. This defines one of the following two implementation requirements for the object,
  - M. This specifies implementation of this object is mandatory.
  - O. This specifies implementation of this object is optional.
5. Access. This defines the access type of the object,
  - N-ACC. This object is not accessible.
  - RW. This object is both readable and writable.

- RO. This object is readable only.
  - RC. This object is readable and the table entry is dynamically created.
6. Comments. This provides a brief description of the object or additional information which is not part of the description in the MIB module.
  7. Apply to. Indicates to whether the object applies to the Access NC and/or the Client Nodes.
  8. The template of the mocaIfConfigTable , mocaIfStatusTable and mocaNotifications objects includes one additional column:  
Mapped to Managed Parameter. Indicates, when applicable, the MoCA Managed Parameter / Event specified in [5] the configuration / notification object is mapped to.

### 9. Node Related Objects

There are seven tables which contain objects specific to the managed interface.

1. Interface Configuration Table
2. Interface Receive Level Added PHY Margin (RLAPM) Table
3. Interface Subcarrier Added PHY Margin (SAPM) Table
4. Interface Alternate Channel Assessment (ACA) Table
5. Interface Status Table
6. Interface Statistics Table
7. Link Statistics Table

### 3.1.1 Interface Configuration Table

The Interface Configuration Table, or mocaIfConfigTable, supports the configuration of RF frequency, transmit power, link privacy, and traps related parameters in the managed MoCA Node. The managed MoCA Node is identified by the interface index (ifIndex).

The support of the configuration table is optional, and all of its objects are writable.

**Table 1: mocaIfConfigTable**

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB		
1	mocaIfAccessPassword	DisplayString	M	RW	Value of the Access Network Password used by the Node	APSWD	Access NC & Client Node
2	mocaIfAuthenticationExempt	OCTET STRING	M	RW	A list that indicates the MAC addresses of the Client Nodes exempted from authentication (see 4.2 in Appendix A).	AUTHENTICATION <sub>EXEMPT</sub>	Access NC

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
3	mocaIfBand	BITS	M	RW	A bitmask defining one or multiple bands of operation of the Node among all the supported bands.	BND	Access NC & Client Node
4	mocaIfChannelMask	BITS	M	RW	A bitmask that defines the supported channels in Network Search.		Client Node
5	mocaIfClientResetReqNodeMask	BITS	M	RW	The value for the RESET_NODE_BITMASK in a Client Reset PIE that the ME wants the Access NC to send.	CLIENT_RESET_REQ_NODEMASK	Access NC
6	mocaIfClientResetReqSend	Unsigned32	M	RW	Request the Access NC to send Client Reset PIEs with the RESET_NODE_BITMASK set to CLIENT_RESET_REQ_NODEMASK	CLIENT_RESET_REQ_SEND	Access NC
7	mocaIfEnable	Boolean	M	RW	Enables or disables the MoCA interface		Access NC & Client Node
8	mocaIfFirstOffset	Integer32	M	RW	The offset of the First Channel's center frequency below the center frequency of the Primary Channel uses when the Node forms the network (valid when NUM_OF_CHANNELS ≥ 3)	FIRST_CHANNEL_OFFSET	Access NC & Client Node
9	mocaIfLof	Unsigned32	M	RW	The value of LOF for the current band of operation of the Node	LOF	Access NC & Client Node
10	mocaIfLofUpdateEnable	Boolean	M	RW	Controls whether the LOF is updated when joining a network	LOF_UPDATE_EN	Access NC & Client Node
13	mocaIfNetworkNameAdmissionRules	INTEGER	M	RW	Indicates to the Node which Admission rules to follow: Join no network (NONE), Join any network (ALL), Join only the network with a network name that matched NETWORK_NAME_NC_NN (NAME)	NETWORK_NAME_ADMIT	Client Node
14	mocaIfNetworkNameNcNn	DisplayString	M	RW	The Network Name the Node is to use for any MoCA network it creates, or indicating no name to use for the network. When NETWORK_NAME_ADMIT = NAME this parameter indicates to the Node the name of the network to join.	NETWORK_NAME_NC_NN	Access NC & Client Node
15	mocaIfNumChannels	Unsigned32	M	RW	The maximum number of 100MHz channels the Access NC uses when it forms the network	NUM_CHANNELS	Access NC



ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
16	mocaIfPer25Mode	INTEGER	M	RW	Controls which PER mode the receiver Node uses for calculating the bitloading of MoCA 2.5 Unicast profiles {NPER (0),VLPER(1)}	PER_2.5 <sub>MODE</sub>	NC & Client Node
17	mocaIfPerCentPriority	OCTET STRING	M	RW	A list of items instructing the Access NC of the the maximum percentage of the wire time allowed to each priority level (see 4.9 in Appendix A).	PRIORITY <sub>PERCENTAGE</sub>	Access NC
18	mocaIfPhyThreshold	Unsigned32	M	RW	The PHY Rate threshold in Mbps used by mocaIfPhyThresholdEnable (for Monitoring/Testing purpose)		Access NC & Client Node
19	mocaIfPowerControlTargetRate	Unsigned32	M	RW	Controls the target PHY Rate of the MoCA Node for TPC (Txm Power Control) algorithm		Access NC & Client Node
20	mocaIfPowerStateRequest	INTEGER	M	RW	Carries a request to the Client Node to transition into a specified Power State.	POWER_STATE <sub>REQ</sub>	Access NC & Client Node
21	mocaIfPrimaryOffset	Integer32	M	RW	The offset of the Primary Channel's center frequency relative to the center frequency of the Beacon Channel uses when the Node forms the network	PRIMARY <sub>OFFSET</sub>	Access NC
22	mocaIfPrivacyEnable	Boolean	M	RW	Determines whether privacy is enabled or disabled	PRIVACY <sub>EN</sub>	Access NC & Client Node
23	mocaIfPrivacyNetAatekRefresh	Unsigned32	M	RW	Dynamic Network AATEK Refresh Period (in mins) (Max: 6 hours)		Access NC
24	mocaIfPrivacyPairwAapmkRefresh	Unsigned32	M	RW	Dynamic Pairwise AAPMK Refresh Period (in mins) (Max: 72 hours)		Access NC
25	mocaIfPrivacyPairwAatekRefresh	Unsigned32	M	RW	Dynamic Pairwise AATEK Refresh Period (in mins) (Max: 6 hours)		Access NC
27	mocaIfRlapmEnable	Boolean	M	RW	Controls whether the RLAPM function is enabled and the Node needs to apply it, or disabled and the Node needs to ignore it.	RLAPM <sub>EN</sub>	Access NC & Client Node
28	mocaIfRlapmProfileSelect	Unsigned32	M	RW	A list of items instructing the Node of the PHY margin to add to the bitloading of all the subcarriers for each received power level (see Section 7.12.2 of [5])		Access NC & Client Node

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
29	mocaIfSapmEnable	Boolean	M	RW	Controls whether the SAPM function is enabled and the Node needs to apply it, or disabled and the Node needs to ignore it.	SAPM <sub>EN</sub>	Access NC & Client Node
30	mocaIfSapmProfileSelect	Unsigned32	M	RW	A list of SAPM values representing the PHY margin the Node is required to add to the bitloading of each of the available subcarriers when the received power level is below ARPL_THLD (see Section 7.12.1 of [5]).		NC & Client Node
31	mocaIfSecondaryOffset	Integer32	M	RW	The offset of the Secondary Channel's center frequency relative to the center frequency of the Primary Channel uses when the Node forms the network	SECONDARY <sub>OFFSET</sub>	Access NC
32	mocaIfTpcEnable	Boolean	M	RW	Selects whether TPC is enabled or disabled	TPC <sub>EN</sub>	Access NC & Client Node
33	mocaIfTpcTargetRateNper	Unsigned32	M	RW	Controls the target PHY Rate of the MoCA Node when TPC is enabled and PER <sub>MODE</sub> = NPER		Access NC & Client Node
34	mocaIfTrafficPermissionEtherType	Unsigned32	M	RW	Specifies the EtherType value of MSDUs that are allowed to transmit/receive to/from a Non-Secured Admitted Client Node	TRAFFIC_PERMISSION <sub>ETHERTYPE</sub>	Access NC & Client Node
35	mocaIfTrapClientResetFailEn	Boolean	M	RW	Enables mocaTrapClientResetFail (see 3.3)	CLIENT_RESET <sub>FAIL</sub>	Access NC & Client Node
36	mocaIfTrapClientResetSuccessEn	Boolean	M	RW	Enables mocaTrapClientResetSuccess (see 3.3)	CLIENT_RESET <sub>SUCCESS</sub>	Access NC & Client Node
37	mocaIfTrapConnectedNodesChangeEn	Boolean	M	RW	Enables mocaTrapConnectedNodesChange (see 3.3)	CONNECTED_NODES <sub>CHG</sub>	Access NC & Client Node
38	mocaIfTrapLmoEn	Boolean	M	RW	Enables 'mocaTrapLmoStatus'		Access NC & Client Node
39	mocaIfTrapNcPrivSupportedRecEn	Boolean	M	RW	Enables mocaTrapNcPrivacySupportedRec (see 3.3)	PRIVACY <sub>REC</sub>	Access NC & Client Node
40	mocaIfTrapPhyThresholdEn	Boolean	M	RW	Enables 'mocaTrapBelowPhyThreshold' and 'mocaTrapAbovePhyThreshold' (see 3.3) (for Monitoring/Testing purpose)		Access NC & Client Node

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
41	mocaIfTrapPowerStateEn	Boolean	M	RW	Enables MoCA Power States traps : mocaTrapPowerStateBcstRec, mocaTrapPowerStateM0Nc, mocaTrapPowerStateResp, mocaTrapPowerStateTrnsReq, mocaTrapPowerStateUcstPen and mocaTrapPowerStateWupUr (see 3.3) (for Monitoring/Testing purpose)	POWER_STATE <sub>BCST_REC</sub> POWER_STATE <sub>M0_NC</sub> POWER_STATE <sub>RESP</sub> POWER_STATE <sub>TRNS_REQ</sub> POWER_STATE <sub>UCST_PEN</sub> POWER_STATE <sub>WUP_UR</sub>	Access NC & Client Node
42	mocaIfTrapStatusChangeEn	Boolean	M	RW	Enables 'mocaTrapIfStatusChange' (see 3.3) (for Monitoring/Testing purpose)		Access NC & Client Node
43	mocaIfTxPowerLimit	Unsigned32	M	RW	Controls the transmit power BACKOFF (in dB) to reduce interferences		Access NC & Client Node
42	mocaIfPowerContour	INTEGER	M	RW	Sets Power Contour used by the Network	POWER_CONTOUR <sub>NC</sub>	Access NC

### 3.1.2 Interface RLAPM Table

The Interface RLAPM Table, or mocaIfRlapmTable, supports the configuration of RLAPM (Receive Level Added PHY Margin) function parameters for the selected profile in the managed MoCA Node, identified by the interface index (ifIndex), the profile number (mocaIfRlapmProfile), and the channel frequency (mocaIfRlapmFrequency).

The support of this table is optional.

**Table 2: mocaIfRlapmTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaIfRlapmProfile	Unsigned32	M	N-ACC	RLAPM (Receive Level Added PHY Margin) profile number	Access NC & Client Node
2	mocaIfRlapmFrequency	Unsigned32	M	N-ACC	MoCA frequency used for the RLAPM (Receive Level Added PHY Margin) table	Access NC & Client Node
3	mocaIfRlapmGarpl	Unsigned32	M	RC	GARPL (Global Aggregated Receive Power Level). Integer in the range of 0 to 65	Access NC & Client Node
4	mocaIfRlapmPhyMargin	Unsigned32	M	RC	PHY Margin values of RLAPM (Receive Level Added PHY Margin). Integer in the range of 0 to 60	Access NC & Client Node

ID	Object Name	Type	Req't	Access	Comments	Apply to
5	mocalfRlapmStatus	RowStatus	M	RC	Control and reflect the status of a row in this table. Creation of a row can either be done with 'CreateAndWait' or 'CreateAndGo'	Access NC & Client Node

### 3.1.3 Interface SAPM Table

The Interface SAPM Table, or mocalfSapmTable, supports the configuration of SAPM (Subcarrier Added PHY Margin) parameters for the selected profile in the managed MoCA Node, identified by the interface index (ifIndex), the profile number (mocalfSapmProfile), and the channel frequency (mocalfSapmFrequency).

The support of this table is optional.

**Table 3: mocalfSapmTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocalfSapmProfile	Unsigned32	M	N-ACC	SAPM (Subcarrier Added PHY Margin) profile number	Access NC & Client Node
2	mocalfSapmFrequency	Unsigned32	M	N-ACC	SAPM (Subcarrier Added PHY Margin) function frequency	Access NC & Client Node
3	mocalfSapmARPLTHLD	Unsigned32	M	RC	SAPM (Subcarrier Added PHY Margin) aggregate Received Power Level Threshold. Integer in the range of 0 to 65 in 1dB steps.	Access NC & Client Node
4	mocalfSapmPhyMargin	OCTET STRING	M	RC	1 Byte Hex array values with no delimiters which indicates PHY Margin values of SAPM (Subcarrier Added PHY Margin) function. Integer in the range of 0 to 120 in 0.5dB steps.	Access NC & Client Node
5	mocalfSapmStatus	RowStatus	M	RC	Control and reflect the status of a row in this table. Creation of a row can either be done with 'CreateAndWait' or 'CreateAndGo'	Access NC & Client Node

### 3.1.4 Interface ACA Table

The Interface mocalfAcaTable, is used by the managed MoCA Node to initiate On Demand LMO. The managed MoCA Node is identified by the interface index (ifIndex).

The support of this table is optional.

**Table 4: mocalfAcaTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

1	mocalfAcaChannel	Unsigned32	M	RW	The channel number under assessment, starting from 0 in increments of 25 MHz	Access NC & Client Node
2	mocalfAcaInitiate	Boolean	M	RW	An ACA (Alternate Channel Assessment) request is initiated when this parameter is written	Access NC & Client Node
3	mocalfAcaNodeID	Unsigned32	M	RW	The Source Node ID.	Access NC & Client Node
4	mocalfAcaPowerProfile	OCTET STRING	M	RO	1 Byte Hex array values with no delimiter	Access NC & Client Node
5	mocalfAcaReportNodeMask	BITS	M	RW	Specifies the MoCA Nodes that are requested to be part of the channel assessment. Setting bits correspond to Node ID's of these MoCA Nodes (LSB corresponds to Node ID 0x0),	Access NC & Client Node
6	mocalfAcaStatus	INTEGER	M	RO	Status: "SUCCESS", "FAIL-BAD CHANNEL", "FAIL-NO EVM PROBE", "FAIL", "IN-PROGRESS"	Access NC & Client Node
7	mocalfAcaStatusTrapCompleted	Boolean	M	RO	Indicated when Power Profile is ready	Access NC & Client Node
8	mocalfAcaTotalRxPower	Integer32	M	RO	dBm (-128 to +127 dBm)	Access NC & Client Node
9	mocalfAcaType	INTEGER	M	RW	The ACA (Alternate Channel Assessment) type is either "EVM" or "QUIET"	Access NC & Client Node

### 3.1.5 Interface Status Table

The Interface Status Table, or 'mocalfStatusTable', provides status information related to the operation of the managed MoCA Node. The managed MoCA Node is identified by the interface index (ifIndex).

The support of this table is mandatory.

**Table 5: mocalfStatusTable**

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB		
1	mocalfAeNumber	Unsigned32	M	RO	Maximum number of allocation elements, excluding the TAUs and the Dummy DAUs, in one MAP the MoCA Node can process		Access NC & Client Node
2	mocalfAggregationSize	Unsigned32	M	RO	Maximum number of bytes in one aggregate which can be received by the MoCA Node		Access NC & Client Node
3	mocalfAuthenticationResult	OCTET STRING	M	RO	The Authentication status of each Node (see 4.3 in Appendix A)	AUTHENTICATIO N_RESULT	Access NC
4	mocalfBwMgmtLinkParameters	OCTET STRING	M	RO	A list of items reporting the PHY rates and the related burst overheads of all the	BW_MGMT_LINK_PAR AMETERS	Access NC

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org) [help@mocalliance.org](mailto:help@mocalliance.org)

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
					links (see 4.4 in Appendix A).		
5	mocalfBwMgmtNetParameters	OCTET STRING	M	RO	A list of items reporting the minimum RR interval and network overhead (see 4.5 in Appendix A)	BW_MGMT <sub>NET_PARAMETERS</sub>	Access NC & Client Node
6	mocalfBwMgmtNodeParameters	OCTET STRING	M	RO	A list of items reporting the maximum aggregation capabilities of all the Nodes in the network (see 4.6 in Appendix A)	BW_MGMT <sub>NODE_PARAMETERS</sub>	Access NC & Client Node
7	mocalfChannelSupport	BITS	M	RO	Specify a list of RF center frequencies represented by a bitmask which this MoCA Node can support to form or join a MoCA network.		Access NC & Client Node
8	mocalfClientResetFailNodeMask	BITS	M	RO	A bitmask that reports the Client Nodes that failed to ACK the Client Reset PIE. A bit set to 1 in the $i^{\text{th}}$ bit position indicates that the Client Node with Node ID = $i$ failed to ACK, where the LSB corresponds to Node ID 0.	CLIENT_RESET <sub>FAILURE_NODEMASK</sub>	Access NC
9	mocalfConnectedNodesInfo	OCTET STRING	M	RO	A list of items reporting the information of each MoCA Access Node of the network (see 4.7 in Appendix A)	CONNECTED_NODE <sub>INFO</sub>	Access NC
10	mocalfFlowInfo	OCTET STRING	M	RO	A list that reports the MAC Addresses of all the Multicast Flows and Unicast Flows. (see 4.8 in Appendix A)	FLOW <sub>INFO</sub>	Access NC & Client Node
11	mocalfLinkDownCount	Counter32	M	RO	The number of times that this Client Node has lost link with the Access NC since the interface was enabled.		Client Node
12	mocalfLinkUpTime	Unsigned32	M	RO	Time in seconds that the MoCA Node has been part of MoCA network		Access NC & Client Node
13	mocalfLmoNodeID	Unsigned32	M	RO	The Node ID of the LMO Node.		Access NC
14	mocalfMacAddress	MacAddress	M	RO	MAC address of the MoCA Node's MoCA interface		Access NC & Client Node
15	mocalfMaxEgressNodeBw	Unsigned32	M	RO	Maximum throughput supported by this Egress Node		Access NC & Client Node
16	mocalfMaxIngressNodeBw	Unsigned32	M	RO	Maximum throughput supported by this Ingress Node		Access NC & Client Node
18	mocalfMocaVersion	INTEGER	M	RO	MoCA version supported by this interface reported in the NODE_PROTOCOL_SUPPORT field of the Admission Request.		Access NC & Client Node

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
19	mocalfName	DisplayString	M	RO	Same as if Name in IF-MIB		Access NC & Client Node
20	mocalfNcPrivacySupported	Unsigned32	O	RO	The latest NC_PRIVACY_SUPPORTED value received from the Access NC. Before any NC_PRIVACY_SUPPORTED is received, reflects the value of the parameter PRIVACY_SUPPORTED.	PRIVACY <sub>NC</sub>	Client Node
21	mocalfNetworkNamePayload	DisplayString	M	RO	The payload of the latest Network Name message received from the Access NC.	NETWORK_NAME PAYLOAD	Client Node
22	mocalfNetworkState	INTEGER	M	RO	Network state of the MoCA Node.		Access NC & Client Node
23	mocalfNetworkVersion	INTEGER	M	RO	MoCA version supported in this network as reported in the MOCA_VERSION field of the Beacon.		Access NC & Client Node
24	mocalfNodeID	Unsigned32	M	RO	Node ID of the MoCA Node		Access NC & Client Node
25	mocalfNumNodes	Unsigned32	M	RO	Number of MoCA Nodes in the network		Access NC
26	mocalfNumPriority	Unsigned32	M	RO	Reports the maximum number of priority levels that the Access NC can support	PRIORITY <sub>NUM</sub>	Client Node
27	mocalfPasswordHash	DisplayString	O	RO	Specify the MoCA password HASH using an ASCII String. See Appendix B. Access is optional when used with SNMP v1 of SNMP v2c protocol		Access NC & Client Node
28	mocalfPduNumber	Unsigned32	M	RO	Maximum number of PDUs in one aggregate which can be received by the MoCA Node		Access NC & Client Node
29	mocalfPfsPrivacyProvision	BITS	M	RO	A list that indicates which Client Nodes have Perfect Forward Secrecy (PFS) with the Access NC.  A bit set to 1 in the i <sup>th</sup> bit position indicates that the Client Node with Node ID = i has PFS with the Access NC, where the LSB corresponds to Node ID 0.	PRIVACY <sub>PFS_PROVISION</sub>	Access NC
30	mocalfPowerStateCap	BITS	M	RO	Reports the Power State capabilities of the Node, i.e. which Power States it can transition to.	POWER_STATE <sub>CAP</sub>	Access NC & Client Node
31	mocalfResetCount	Counter32	M	RO	Count of MoCA resets since the interface was enabled for this Node.		Access NC & Client Node

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter	Apply to
32	mocalfResetReason	DisplayString	M	RO	Provides a description for reset link, or MoCA Node join failure.		Access NC & Client Node
33	mocalfRFChannel	Unsigned32	M	RO	RF channel number of the MoCA Node.		Access NC & Client Node
34	mocalfSoftwareVersion	OCTET STRING	M	RO	Aligned to the Certificate information Up to 82 characters. May report additional vendor specific information. See 4.11		Access NC & Client Node
35	mocalfStatus	INTEGER	M	RO	Indicate the current status of the MoCA interface with value =1 is disable, 2 is no link, and 3 is link-up		Access NC & Client Node
36	mocalfSupportedBands	BITS	M	RO	All the supported bands and sub-bands by the MoCA Node		Access NC & Client Node
37	mocalfTxGcdRate	Unsigned32	M	RO	Tx GCD rate of the Access NC Node		Access NC
38	mocalfPrivacySupported	BITS	O	RO	Privacy modes supported when PRIVACY <sub>EN</sub> = ENABLED	PRIVACY <sub>SUPPORTED</sub>	Access NC & Client Node
39	mocalfLinkState	OCTET STRING	M	RO	This object is meaningful only after a Link Failure to indicate the state of the network. See 4.12.	<ul style="list-style-type: none"> <li>LINK_STATE_IL<sub>ST</sub></li> <li>ACF_TYPE<sub>LST</sub></li> </ul>	Access NC & Client Node
40	mocalfBeaconTxPwr	Integer32	M	RO	Transmit Power in dBm of Beacon reported in the BEACON_TRANSMIT_POWER field of the beacon.	<ul style="list-style-type: none"> <li>BEACON<sub>TX_PWR</sub></li> </ul>	Access NC & Client Node
41	mocalfBeaconRxPwr	Integer32	M	RO	Receive Power in dBm of the Beacon Beacon channel Attenuation in dB between the Access NC and the Client Node could be calculated as follows: Atten=mocalfBeaconTxPwr - mocalfBeaconRxPwr	<ul style="list-style-type: none"> <li>BEACON<sub>RX_PWR</sub></li> </ul>	Client Node
42	mocalfNcPowerContour	INTEGER	M	RO	Power Contour reported by the Access NC	<ul style="list-style-type: none"> <li>POWER_CONTO<sub>UR<sub>NC</sub></sub></li> </ul>	Client Node
43	mocalfNcVersion	Unsigned32	M	RO	MoCA Version of the Access NC as reported in the NODE_PROTOCOL_SUPPORT field of the Admission Response.	<ul style="list-style-type: none"> <li></li> </ul>	Access NC & Client Node



### 3.1.6 Interface Statistics Table

The Interface Statistics Table, or 'mocalfStatsTable', provides statistics information on counters for transmission and reception of packets in a managed MoCA Node. The managed MoCA Node is identified by the interface index (ifIndex) and the Priority Index (mocalfPriorityIndex).

The support of this table is mandatory.

**Table 6: mocalfStatsTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocalfRxBroadcast	Counter64	M	RO	Provided in the IF MIB ifOutBroadcastPkts. Indicate the number of Broadcast packets received by this MoCA interface.	Client Node
2	mocalfRxBytes	Counter64	M	RO	Provided in the IF MIB ifOutOctets. Indicate the number of bytes received by this MoCA interface.	Access NC & Client Node
3	mocalfRxCorrectedErrors	Counter64	M	RO	Provided in the IF MIB ifInErrors	Access NC & Client Node
4	mocalfRxDrops	Counter32	M	RO	Number of dropped packets by the receiver of this MoCA interface	Access NC & Client Node
5	mocalfRxMulticast	Counter64	M	RO	Provided in the IF MIB ifOutMulticastPkts. Indicate the number of Multicast packets received by this MoCA interface.	Access NC & Client Node
6	mocalfRxPackets	Counter64	M	RO	Provided in the IF MIB ifInUcastPkts	Access NC & Client Node
7	mocalfRxUnicast	Counter64	M	RO	Provided in the IF MIB ifOutUcastPkts. Indicate the number of Unicast packets received by this MoCA interface	Access NC & Client Node
8	mocalfTxBroadcast	Counter64	M	RO	Provided in the IF MIB ifInBroadcastPkts. Indicate the number of Broadcast packets transmitted by this MoCA interface.	Access NC
10	mocalfTxBytes	Counter64	M	RO	Provided in the IF MIB ifInOctets. Indicate the number of bytes transmitted by this MoCA interface.	Access NC & Client Node
11	mocalfTxDrops	Counter32	M	RO	Provided in the IF MIB ifOutDiscards. Indicate the number of Ethernet packet requests made at the MAC_SAP, destined to the associated Node, and not transmitted by this MoCA interface including drops due to required throttling as required by the specification.	Access NC & Client Node
12	mocalfTxMulticast	Counter64	M	RO	Provided in the IF MIB ifInMulticastPkts. Indicate the number of Multicast packets transmitted by this MoCA interface.	Access NC & Client Node
13	mocalfTxPackets	Counter64	M	RO	Provided in the IF MIB ifOutUcastPkts.	Access NC & Client Node

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

ID	Object Name	Type	Req't	Access	Comments	Apply to
14	mocaIfTxUnicast	Counter64	M	RO	Provided in the IF MIB ifInUcastPkts. Indicate the number of Unicast packets transmitted by this MoCA interface.	Access NC & Client Node

### 3.1.7 Link Statistics Table

This table provides link statistics objects reported by the managed interface for the links to other MoCA Node(s) this MoCA interface is connected to in the MoCA Access network.

This table contains an entry for every MoCA Node in the MoCA Access network this MoCA interface is connected to, excluding this MoCA interface itself. This table is indexed by 'ifIndex' and 'mocaLinkNodeIndex'. Entries in this table cannot be created or deleted by the network management system. All entries are created or deleted by the device software.

The support of this table is optional.

**Table 7: mocaLinkStatsTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaLinkNodeIndex	Unsigned32	M	N-ACC	The Node ID is used as an index to the MoCA Node link statistics table. For Client Node, this index must be 0.	Access NC & Client Node
2	mocaLinkTxDrops	Counter64	M	RO	Provided in the IF MIB ifOutDiscards. Indicate the number of Ethernet packet transmission requests made at the MAC_SAP, destined to the associated Node, and not transmitted by this MoCA interface.	Access NC & Client Node
3	mocaLinkTxPackets	Counter64	M	RO	Provided in the IF MIB ifOutUcastPkts. Indicate the number of unicast Ethernet packets transmitted by this MoCA interface to the associated Node.	Access NC & Client Node

## 3.2 Network Related Objects

There are four tables which contain objects reported by the managed interface for other moCA Nodes in the MoCA network. Note that the P2mp, Link Modulation Tables contain entries for the managed MoCA Node as well.

1. Node Table
2. P2mp Table
3. Link Modulation Table

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

### 3.2.1 Node Table

The MoCA Node Table, or 'mocaNodeTable', is used by the managed MoCA Node to report information about other MoCA Nodes in the MoCA network. This table is indexed by the interface index (ifIndex) and MoCA Node ID (mocaNodeIndex). The managed MoCA Node does not report itself, and hence the MoCA Node ID of the managed MoCA Node does not have an entry in this table. All other MoCA Nodes that the managed MoCA Node has both transmission and reception usable channels with (as indicated by CHANNEL\_USABLE field of the Type 1 Probe Report) have an entry in this table.

The support of this table is mandatory.

**Table 8: mocaNodeTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaNodeIndex	Unsigned32	M	N-ACC	The Node ID is used as an index to the MoCA Node table	Access NC
2	mocaNodeAeNumber	Unsigned32	M	RO	Maximum number of allocation elements, excluding the TAUs and the Dummy DAUs, in one MAP the MoCA Node identify by the mocaNodeIndex can process	Access NC
3	mocaNodeAggregationSize	Unsigned32	M	RO	Maximum number of bytes in one aggregate which can be received by the MoCA Node identified by mocaNodeIndex	Access NC
4	mocaNodeMacAddress	MacAddress	M	RO	Indicate the MAC address of the MoCA Node identified by 'mocaNodeIndex'	Access NC
5	mocaNodeMocaVersion	INTEGER	M	RO	Indicate the MoCA version supported by the MoCA Node identified by 'mocaNodeIndex' as reported in the NODE_PROTOCOL_SUPPORT field of the AdmissionResponse.	Access NC
6	mocaNodePduNumber	Unsigned32	M	RO	Maximum number of PDUs in one aggregate which can be received by the MoCA Node	Access NC
7	mocaNodePowerState	INTEGER	M	RO	Power State of the MoCA Node identify by the 'mocaNodeIndex'	Access NC
8	mocaNodeRxCorrected	Counter64	M	RO	Number of received Ethernet packets that have errors and are corrected by the managed MoCA Node from the MoCA Node identified by 'mocaNodeIndex'	Access NC & Client Nodes
9	mocaNodeRxDrops	Counter32	M	RO	The number of scheduled MoCA frames that are not detected or have uncorrectable errors and are dropped by the managed MoCA Node from the MoCA Node identified by 'mocaNodeIndex'	Access NC & Client Nodes
10	mocaNodeRxPackets	Counter64	M	RO	Number of good Ethernet packets received by the managed MoCA Node from the MoCA Node identified by 'mocaNodeIndex', includes unicast, multicast, and broadcast packets	Access NC & Client Nodes
14	mocaNodePDelay	Integer32	M	RO	Propagation delay measured in units of 10ns between the local Node and the Node identified by mocaNodeIndex	Access NC

### 3.2.2 P2mp Table

The P2mp Table, or 'mocaP2mpTable', provides the PHY rates between the Access NC and every Client Node in the MoCA Access network. Each PHY rate is associated with an interface index (ifIndex), transmit Node ID (mocaP2mpTableTxNodeIndex) and receive Node ID (mocaP2mpTableRxNodeIndex). PHY rate is the transmit data rate from the MoCA Node identified by the transmit Node ID, to the MoCA Node identified by the receive Node ID.

The support of this table is mandatory.

**Table 9: mocaP2mpTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaP2mpPhyRates	OCTET_STRING	M	O	The transmit and received PHY rate (in Mbps) for each Access NC to Client Node link (see 4.10 in Appendix A)	Access NC

### 3.2.3 Link Modulation Table

The mocaScModTable contains the subcarrier modulation for every link in the MoCA Access network. The mocaScModTable is indexed by 'ifIndex', 'mocaTxNodeId', and 'mocaRxNodeId'. Each entry of the table provides the subcarrier modulation for the link from the Node whose Node ID = 'mocaTxNodeId' to the Node whose Node ID = 'mocaRxNodeId'. If 'mocaTxNodeId' = 0 and 'mocaRxNodeId' = 0, the entry contains the GCD subcarrier modulation from the Access NC to all CPE Nodes in the network. Since a MoCA Access network can have less than 63 CPE Nodes, some values of 'mocaTxNodeId' and 'mocaRxNodeId' in the range of 1 to 63 may not exist. Additionally, either mocaTxNodeId or mocaRxNodeId must be 0. Entries in this table cannot be created or deleted by the network management system. All entries are created or deleted by the device software.

The support of this table is optional.

**Table 10: mocaScModTable**

ID	Object Name	Type	Req't	Access	Comments	Apply to
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaTxNodeId	Unsigned32	M	N-ACC	The Node ID of the transmitting MoCA Node	Access NC & Client Node
2	mocaRxNodeId	Unsigned32	M	N-ACC	The Node ID of the receiving MoCA Node	Access NC & Client Node
3	mocaLinkType	INTEGER	M	RO	Indicates the type of link which exists from the node with Node ID = mocaTxNodeId to the node with Node ID = mocaRxNodeId.. The link types are: Primary, Bonded, 300 MHz MoCA 2.5 PHY, 400 MHz MoCA 2.5 PHY, or 500 MHz MoCA 2.5 PHY.	Access NC & Client Node

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

ID	Object Name	Type	Req't	Access	Comments	Apply to
4	mocaScModList	OCTET STRING	M	RO	1 Byte Hex array values with no delimiters which indicates the subcarrier modulation for all the subcarriers from the MoCA Node identified by 'mocaTxNodeId' to the MoCA Node identified by 'mocaRxNodeId' (see 4.1.2 in Appendix A).	Access NC & Client Node
5	mocaNodePower	OCTET STRING	M	RO	If 'mocaTxNodeId' is equal to 'mocaIfNodeID', then indicates the unicast transmit power in dBm to the 'mocaRxNodeId'  Else indicates the unicast receive power in dBm from the 'mocaTxNodeId' (see 4.13 in Appendix A).	Access NC & Client Nodes
6	mocaNodeRxSNR	OCTET STRING	M	RO	Average Receive SNR across all available subcarriers based on the EVM probe from the 'mocaTxNodeId' to the 'mocaRxNodeId'. Object does not exist for transmit direction. (see 4.14 in Appendix A).	Access NC & Client Nodes
7	mocaNodePowerReduction	OCTET STRING	M	RO	Power Control back-off in dB used for unicast transmissions from the 'mocaTxNodeId' to the 'mocaRxNodeId' (see 4.15 in Appendix A).	Access NC & Client Nodes

### 3.3 Notification Events

The asynchronous notification events or traps defined in the following table may be sent from the MoCA device to the network management system (NMS). A trap can contain any number of variable bindings (VarBind) or MIB object-value pairs. The traps defined in this document must include the interface index (ifIndex) as one of the VarBind, or as part of the instance of the VarBind.

The support of this table is optional.

**Table 11: mocaNotifications**

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter (Event)
	ifIndex	Integer32	M	N-ACC	From IF-MIB	
1	mocaTrapAbovePhyThreshold	Unsigned32	M	RO	Contain 1 VarBind for 'mocaP2mpTxRate' - (for Monitoring/Testing purpose)	
2	mocaTrapBelowPhyThreshold	Unsigned32	M	RO	Contain 1 VarBind for 'mocaP2mpTxRate' - (for Monitoring/Testing purpose)	
3	mocaTrapClientResetFail	Unsigned32	M	RO	The Access NC reports that at least one Client Node set in the CLIENT_RESET <sub>REQ_NODEMASK</sub> failed to ACK the Client Reset PIE and the node bitmask of the Client Nodes that failed to ACK	CLIENT_RESET <sub>FAIL</sub>

ID	Object Name	Type	Req't	Access	Comments	Mapped to Managed Parameter (Event)
					is available in the parameter CLIENT_RESET <sub>FAIL_NODEMASK</sub>	
4	mocaTrapClientResetSuccess	Unsigned32	M	RO	The Access NC reports that all the Client Nodes specified in the CLIENT_RESET <sub>REQ_NODEMASK</sub> have successfully ACKed the Client Reset Request PIE.	CLIENT_RESET <sub>SUCCESS</sub>
5	mocaTrapConnectedNodesChange	Unsigned32	M	RO	Reports that a New Node has joined the network or an EN has been dropped from the network	CONNECTED_NODES <sub>CHG</sub>
6	mocaTrapIfStatusChange	Unsigned32	O	RO	Contain 1 Varbind for 'mocaIfStatus' (for Monitoring/Testing purpose)	
7	mocaTrapLmoStatus	Unsigned32	O	RO	Reports when the MoCA Node is starting LMO (for Testing purpose)	
8	mocaTrapNcPrivacySupportedRec	Unsigned32	M	RO	Reports that the Node has received new NC_PRIVACY_SUPPORTED value which is available through the parameter PRIVACY <sub>NC</sub> .	PRIVACY <sub>REC</sub>
9	mocaTrapPowerStateBcstRec	Unsigned32	M	RO	When the Node is in Power State M1 or M2, reports that the Node has received a Broadcast data MSDU, which is available at the data interface.	POWER_STATE <sub>BCST_REC</sub>
10	mocaTrapPowerStateM0Nc	Unsigned32	M	RO	When a Node is in Power State M1 reports that it is going to move to Power State M0 due to the NC's instruction.	POWER_STATE <sub>M0_NC</sub>
11	mocaTrapPowerStateResp	Unsigned32	M	RO	Carries the response of the Node to a request (through POWER_STATE <sub>REQ</sub> ) to transition into a desired Power State.	POWER_STATE <sub>RESP</sub>
12	mocaTrapPowerStateTrnsReq	Unsigned32	M	RO	When a Node is in Power State M1 or M2, requests to transition to Power State M0	POWER_STATE <sub>TRNS_REQ</sub>
13	mocaTrapPowerStateUcstPen	Unsigned32	M	RO	When the Node is in Power State M2, reports that a Unicast data MSDU destined to the Node is pending.	POWER_STATE <sub>UCST_PEN</sub>
14	mocaTrapPowerStateWupUr	Unsigned32	M	RO	When the Node is in Power State M2, reports that a wakeup request from the Access NC due to unspecified reasons.	POWER_STATE <sub>WUP_UR</sub>
15	mocaTrapLinkFailure	Unsigned32	M	RO	Indicate a link failure. If the failure was an Admission or LMO failure, mocaLinkState indicates in which link state the link was dropped	LINK <sub>DROP</sub>

## 4 Appendix A

This appendix provides the encoding for the following objects and data types:

### 4.1 Aca Power Profile

- Object: mocaIfAcaPowerProfile
- Data Type: MocaScMod

#### 4.1.1 Encoding for the Object mocaIfAcaPowerProfile

Below is a real-world example of a reply for mocaIfAcaPowerProfile in Table 5 mocaIfAcaTable:

```

EDD9DADFE5E6E6E7E8E7E8E8E7E7E8E8E7E8E8E7E7E8E6E8E8E8E8E8E7E6E8E7E8E8E7
E8E6E7E8E7E8E7E8E8E8E8E6E8E8E6E7E7E7E8E8E7E7E7E7E7E7E7E7E7E7E8E7E7E9E7E7
E8E8E8E7E7E8E7E8E7E6E8E8E7E8E8E8E8E8E7E8E8E8E7E9E8E9E8E8E8E8E8E9E8E7E8
E8E8E8E8E8E8E8DFDFDADF8DED7DBD6D6D4DCD3D9D2CFCECECDCECDCCCCCCCCCCCC
CCCBCBCBCCBCBCCBCBCBCACBCACBCAC9CBCBCACAC9C9C9CACACACACAC9C9C9C9CAC9C9
C9C9C9CAC8C9C8C9C9C9C9C9C9C9C9C8C8C9C8C9C9C9C8C9C9C9C8C9C9C9C8C9C9C9C8C8
C8C8C9C9C8C9C9C8C9C9C9C9C9C9C9C9C8C8C8C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C8
C8C9C8C8C8C8C8C8C7C8C9C9C9C8C8C9C8C8C9C8C9C9C8C9C8C9C9C9C9C9C9C9C9C9C9C9
C8C9C8C8C8C9C8C8C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9C9
CACACACACAC9CBCACACACACACBCACACACBCBCBCBCBCBCBCBCBCBCBCBCBCBCBCBCBCBCBC
CCCCCCCCDCDCDCDCDCDCDCCECECECECECECECFD0D0D1D2D9D4DCD5D5D7DBD7DEDADDFDBDF
E0E8E8E8E9E8E9E9E8E9E8E8E9E8E9E9E9E8E9E8EAE9E8E9E9E8E9E9E8E9E9E9E8E9E9
E8E9E8E8E8E9E8E9E7E8E8E8E9E9E8E9E8E9E8E9E8E9E8E9E8E9E9E9E8E8E9E9E8E7E8
E9E8E9E8E9E8E8E8E7E8E9E8E8E8E8E9E7E9E7E8E8E9E8E9E7E8E8E8E8E7E7E8E9E9E7
E7E7E7E7E5E0DADA
    
```

The above message can be decoded as shown below. The subcarriers 0 to 511 have values as shown. The SNMP’s TLV header has the number of subcarriers.

Subcarrier	0	1	2	3	4	5	6	7	8	9
0	-19 dB	-39 dB	-38 dB	-33 dB	-27 dB	-26 dB	-26 dB	-25 dB	-24 dB	-25 dB
10	-24 dB	-24 dB	-25 dB	-25 dB	-24 dB	-24 dB	-25 dB	-24 dB	-24 dB	-25 dB
20	-25 dB	-24 dB	-26 dB	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB	-25 dB
30	-26 dB	-24 dB	-25 dB	-24 dB	-24 dB	-25 dB	-24 dB	-26 dB	-25 dB	-24 dB
40	-25 dB	-24 dB	-25 dB	-24 dB	-24 dB	-24 dB	-24 dB	-26 dB	-24 dB	-24 dB
50	-26 dB	-25 dB	-25 dB	-25 dB	-24 dB	-24 dB	-25 dB	-25 dB	-25 dB	-25 dB

60	-25 dB	-25 dB	-25 dB	-25 dB	-25 dB	-25 dB	-24 dB	-25 dB	-25 dB	-23 dB
70	-25 dB	-25 dB	-24 dB	-24 dB	-24 dB	-25 dB	-25 dB	-24 dB	-25 dB	-24 dB
80	-25 dB	-26 dB	-24 dB	-24 dB	-25 dB	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB
90	-25 dB	-24 dB	-24 dB	-24 dB	-25 dB	-23 dB	-24 dB	-23 dB	-24 dB	-24 dB
100	-24 dB	-24 dB	-24 dB	-24 dB	-23 dB	-24 dB	-25 dB	-24 dB	-24 dB	-24 dB
110	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB	-33 dB	-33 dB	-38 dB	-33 dB
120	-40 dB	-34 dB	-41 dB	-37 dB	-42 dB	-42 dB	-44 dB	-36 dB	-45 dB	-39 dB
130	-46 dB	-49 dB	-50 dB	-50 dB	-51 dB	-51 dB	-50 dB	-51 dB	-52 dB	-52 dB
140	-52 dB	-52 dB	-52 dB	-52 dB	-52 dB	-53 dB	-53 dB	-53 dB	-52 dB	-53 dB
150	-53 dB	-52 dB	-53 dB	-53 dB	-53 dB	-54 dB	-53 dB	-54 dB	-53 dB	-54 dB
160	-55 dB	-53 dB	-53 dB	-54 dB	-54 dB	-55 dB	-55 dB	-55 dB	-54 dB	-54 dB
170	-54 dB	-54 dB	-54 dB	-55 dB	-55 dB	-55 dB	-55 dB	-54 dB	-55 dB	-55 dB
180	-55 dB	-55 dB	-55 dB	-54 dB	-56 dB	-55 dB	-56 dB	-55 dB	-55 dB	-55 dB
190	-55 dB	-55 dB	-55 dB	-55 dB	-56 dB	-54 dB	-56 dB	-55 dB	-56 dB	-55 dB
200	-55 dB	-56 dB	-54 dB	-55 dB	-55 dB	-56 dB	-56 dB	-55 dB	-55 dB	-56 dB
210	-56 dB	-54 dB	-56 dB	-56 dB	-56 dB	-56 dB	-56 dB	-56 dB	-55 dB	-55 dB
220	-56 dB	-55 dB	-55 dB	-56 dB	-55 dB	-55 dB	-55 dB	-55 dB	-56 dB	-55 dB
230	-55 dB	-55 dB	-55 dB	-56 dB	-56 dB	-56 dB	-55 dB	-55 dB	-55 dB	-55 dB
240	-55 dB	-55 dB	-55 dB	-56 dB	-55 dB	-55 dB	-55 dB	-56 dB	-57 dB	-56 dB
250	-55 dB	-57 dB	-56 dB	-55 dB	-56 dB	-56 dB	-56 dB	-56 dB	-56 dB	-56 dB
260	-56 dB	-57 dB	-56 dB	-55 dB	-55 dB	-55 dB	-56 dB	-56 dB	-55 dB	-56 dB
270	-56 dB	-55 dB	-56 dB	-55 dB	-55 dB	-56 dB	-55 dB	-56 dB	-55 dB	-55 dB
280	-56 dB	-55 dB	-54 dB	-55 dB	-55 dB	-56 dB	-56 dB	-55 dB	-56 dB	-55 dB
290	-56 dB	-54 dB	-56 dB	-56 dB	-55 dB	-56 dB	-56 dB	-55 dB	-55 dB	-54 dB
300	-56 dB	-55 dB	-55 dB	-54 dB	-54 dB	-54 dB	-55 dB	-55 dB	-55 dB	-55 dB
310	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-55 dB	-53 dB	-55 dB	-55 dB
320	-55 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-55 dB



<b>330</b>	-53 dB	-54 dB	-54 dB	-54 dB	-54 dB	-54 dB	-53 dB	-54 dB	-54 dB	-54 dB
<b>340</b>	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB
<b>350</b>	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-53 dB	-52 dB	-52 dB	-52 dB	-51 dB
<b>360</b>	-52 dB	-52 dB	-52 dB	-52 dB	-51 dB	-51 dB	-51 dB	-50 dB	-51 dB	-51 dB
<b>370</b>	-51 dB	-50 dB	-50 dB	-50 dB	-50 dB	-50 dB	-50 dB	-50 dB	-49 dB	-48 dB
<b>380</b>	-48 dB	-47 dB	-46 dB	-39 dB	-44 dB	-36 dB	-43 dB	-43 dB	-41 dB	-37 dB
<b>390</b>	-41 dB	-34 dB	-38 dB	-33 dB	-37 dB	-33 dB	-32 dB	-24 dB	-24 dB	-24 dB
<b>400</b>	-23 dB	-24 dB	-23 dB	-23 dB	-24 dB	-23 dB	-24 dB	-24 dB	-23 dB	-24 dB
<b>410</b>	-23 dB	-23 dB	-23 dB	-24 dB	-23 dB	-24 dB	-22 dB	-23 dB	-24 dB	-23 dB
<b>420</b>	-23 dB	-24 dB	-23 dB	-23 dB	-24 dB	-23 dB	-23 dB	-23 dB	-23 dB	-24 dB
<b>430</b>	-23 dB	-23 dB	-24 dB	-23 dB	-24 dB	-24 dB	-24 dB	-23 dB	-24 dB	-23 dB
<b>440</b>	-25 dB	-24 dB	-24 dB	-24 dB	-23 dB	-23 dB	-24 dB	-23 dB	-24 dB	-23 dB
<b>450</b>	-24 dB	-23 dB	-24 dB	-23 dB	-24 dB	-23 dB	-24 dB	-23 dB	-23 dB	-23 dB
<b>460</b>	-23 dB	-24 dB	-24 dB	-23 dB	-23 dB	-24 dB	-25 dB	-24 dB	-23 dB	-24 dB
<b>470</b>	-23 dB	-24 dB	-23 dB	-24 dB	-24 dB	-24 dB	-25 dB	-24 dB	-23 dB	-24 dB
<b>480</b>	-24 dB	-24 dB	-24 dB	-23 dB	-25 dB	-23 dB	-25 dB	-24 dB	-24 dB	-23 dB
<b>490</b>	-24 dB	-23 dB	-25 dB	-24 dB	-24 dB	-24 dB	-24 dB	-24 dB	-25 dB	-25 dB
<b>500</b>	-24 dB	-23 dB	-23 dB	-25 dB	-25 dB	-25 dB	-25 dB	-25 dB	-27 dB	-32 dB
<b>510</b>	-38 dB	-38 dB								

#### 4.1.2 Encoding for the Data Type MocaScModList

The MocaScModList type is an OCTET STRING and represents a string of individual bytes with no delimiters.

Each byte in the object represents the modulation of one subcarrier. The value of each byte is interpreted as shown in the following table.

Byte Value	Subcarrier Modulation
0	Not used
1	BPSK
2	QPSK

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

3	8 QAM
4	16 QAM
5	32 QAM
6	64 QAM
7	128 QAM
8	256 QAM
9	512 QAM
10	1024 QAM

Subcarriers are encoded in sequential bytes starting from the most significant byte to the lowest significant byte. The lowest significant byte carries the subcarrier modulation for the lowest frequency subcarrier and the most significant byte carries the subcarrier modulation for the highest frequency subcarrier.

If 'mocaTxNodeId' = 0 and 'mocaRxNodeId' = 0, the subcarrier modulation encoded in this array is for the GCD channel from the Access NC to the CPE nodes.

For Primary Channel, Bonded Channel, 300 MHz MoCA 2.5 PHY, 400 MHz MoCA 2.5 PHY, and 500 MHz MoCA 2.5 PHY channels the number of bytes in this object is 512, 1024, 1536, 2048, and 2560 respectively.

## 4.2 Authentication Exempt

- Object: mocaIfAuthenticationExempt
- Data Type: mocaAuthenticationExempt

### 1. Encoding for the Data Type mocaAuthenticationExempt

OCTET STRING of list of Node GUID MAC addresses

NUM_ADDRS	4 bytes	Number of addresses in the list
{		
GUID	8 bytes	Node GUID MAC address
} [NUM_ADDRS]		

## 4.3 Authentication Result

- Object: mocaIfAuthenticationResult
- Data Type: mocaAuthenticationResult

### 1. Encoding for the Data Type mocaAuthenticationResult

OCTET STRING of Client Node ID vector

Authenticated Client Node ID Vector	8 bytes	Bit value in the ith bit position indicates that Node ID = (i) is: - successfully authenticated (1) - unauthenticated (0)  Bit 0 (Access NC Node ID) is always set
-------------------------------------	---------	--

#### 4.4 Link Parameters

- Object: mocaIfBwMgmtLinkParameters
- Data Type: mocaBwMgmtLinkParameters

##### 1. Encoding for the Data Type mocaBwMgmtLinkParameters

OCTED STRING of Client Node ID vector followed by an enumeration of {PHYRate, BurstOverhead} items.

Client Node ID Vector	8 bytes	A bitmask that reports the Node IDs of Client Nodes where the LSB corresponds to Node ID 0. For each Node ID position set to 1 starting from the LSB to the MSB, there is an enumeration of PHY Rates and Burst Overheads in the fields below.  Bit 0 (Access NC Node ID) must not be set
NUM_NODES	4 bytes	Number of Node IDs indicated in the Client Node ID Vector
{		
Uplink PHY Rate	2 bytes	in Mbps
Uplink Burst Overhead	2 bytes	in $\mu$ S
Downlink PHY Rate	2 bytes	in Mbps
Downlink Burst Overhead	2 bytes	in $\mu$ S
} [ NUM_NODES ]		

#### 4.5 Net Parameters

- Object: mocaIfBwMgmtNetParameters
- Data Type: mocaBwMgmtNetParameters

##### 1. Encoding for the Data Type mocaBwMgmtNetParameters

OCTET STRING of {MinRRInterval, NetworkOverhead}

MinRRInterval	2 Bytes	in $\mu$ S
NetworkOverhead	1 Byte	in percentage [0...100]

#### 4.6 Node Parameters

- Object: mocaIfBwMgmtNodeParameters
- Data Type: mocaBwMgmtNodeParameters

##### 1. Encoding for the Data Type mocaBwMgmtNodeParameters

OCTET STRING of Node ID vector followed by an enumeration of {MaxBytesAggregation, MaxPacketsAggregation} items

Node ID Vector	8 bytes	A bitmask that reports the Node IDs of Client Nodes where the LSB corresponds to Node ID 0. For each Node ID position set to 1 starting from the LSB to the MSB, there is an
----------------	---------	--

		enumeration of Max Bytes & Max Packets Aggregations in the fields below.
NUM_NODES	4 bytes	Number of Node IDs indicated in the Node ID Vector
{		
MaxBytesAggregation	4 bytes	in bytes
MaxPacketsAggregation	4 bytes	in number of packets
}	[NUM_NODES]	

#### 4.7 Connected Node Info

- Object: mocaIfConnectedNodesInfo
- Data Type: mocaConnectedNodesInfo

##### 1. Encoding for the Data Type mocaConnectedNodesInfo

OCTET STRING of Node ID vector followed an enumeration of {GUID, MoCA Version Number} items

Node ID Vector	8 bytes	A bitmask that reports the Node IDs of Client Nodes where the LSB corresponds to Node ID 0. For each Node ID position set to 1 starting from the LSB to the MSB, there is an enumeration of GUID MAC addresses & MoCA versions in the fields below.
NUM_NODES	4 bytes	Number of Node IDs indicated in the Client Node ID Vector
{		
GUID	8 bytes	Node GUID MAC Address
MoCA Version	1 byte	MoCA version
}	[NUM_NODES]	

#### 4.8 Flow Info

- Object: mocaIfFlowInfo
- Data Type: mocaFlowInfo

##### 1. Encoding for the Data Type mocaFlowInfo

OCTET STRING of a list of {MAC Addresses}

NUM_ADDRS	4 bytes	Number of addresses in the list
{		
MAC Address	6 bytes	MAC address
}	[NUM_ADDRS]	

#### 4.9 PerCent Priority

- Object: mocaIfPerCentPriority
- Data Type: mocaPerCentPriority

##### 1. Encoding for the Data Type mocaPerCentPriority

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

OCTET STRING of enumeration of {Uplink Percentage, Downlink Percentage} items in incrementing Priority level (from 0 to 7 max)

MAX_PRI	1 byte	Max Priority Level in the enumeration [0...7]
{		
Uplink Percentage	1 byte	in percentage: [0...100]
Downlink Percentage	1 byte	in percentage: [0...100]
} [MAX_PRI]		

#### 4.10 P2mpPhyRates

- Object: mocaP2mpPhyRates
- Data Type: mocaP2mpNodePhyRate

##### 1. Encoding for the Data Type mocaP2mpNodePhyRate

OCTET STRING of Node ID vector followed by an enumeration of {Tx PHYRate, Rx PHYRate} items. Bit 0 (Access NC Node ID) in the Node ID Vector is always set and the first item indicates the GCD bitloading from the Access NC to the Client Nodes.

Node ID Vector	8 bytes	A bitmask that reports the Node IDs of Nodes where the LSB corresponds to Node ID 0. For each Node ID position set to 1 starting from the LSB to the MSB, there is an enumeration of PHY Rates in the fields below.  Bit 0 (Access NC Node ID) is always set and the first pair in the fields below indicates the txgcd bitloading from the Access NC to the Client Nodes
NUM_NODES	1 byte	Number of Node IDs indicated in the Node ID Vector
{		
Tx PHY Rate	2 bytes	in Mbps
Rx PHY Rate	2 bytes	in Mbps
} [ NUM_NODES]		

#### 4.11 Software Version

- Object: mocaIfSoftwareVersion
- Data Type: mocaSoftwareVersion

Encoding for the Data Type mocaSoftwareVersion:  
OCTET STRING of MoCA\_VendorID and Information field

MoCA Vendor ID	2 bytes	MoCA Vendor ID (allocated by the MoCA Alliance) [6]
----------------	---------	---

Information	80 bytes	Up to 80 characters.
-------------	----------	----------------------

## 4.12 Link State

- Object: mocaIfLinkState
- Data Type: mocaLinkState

Encoding for the Data Type mocaLinkState:

OCTET STRING of LINK\_STATE and ACF\_TYPE

LINK_STATE	1 Byte	LINK_STATE_II from the transmitted/received MAP at the time of the Link Failure.
ACF_TYPE	1 Byte	ACF_TYPE from the transmitted/received Beacon at the time of the Link Failure.

## 4.13 Node Power

- Object: mocaNodePower
- Data Type: mocaPower

Encoding for the Data Type mocaNodePower:

OCTET STRING of per channel list of Tx or Rx Power levels

NUM_VALUES	1 Byte	Function of mocaLinkType: Primary – 1 Bonded – 2 300 MHz MoCA 2.5 PHY – 3 400 MHz MoCA 2.5 PHY – 4 500 MHz MoCA 2.5 PHY – 5
{		
POWER	1 Byte	Transmit or Receive Power Level in dBm depending on direction of Link.
} [NUM_VALUES]		

## 4.14 Node RX SNR

- Object: mocaNodeRxSNR
- Data Type: mocaRxSNR

Encoding for the Data Type mocaNodeRxSNR:

OCTET STRING of per channel list of SNR

NUM_VALUES	1 Byte	Function of mocaLinkType: Primary – 1 Bonded – 2 300 MHz MoCA 2.5 PHY – 3 400 MHz MoCA 2.5 PHY – 4 500 MHz MoCA 2.5 PHY – 5
{		
RX_SNR	1 Byte	Average Receive SNR in dB from last EVM probe.

} [NUM_VALUES]
----------------

## 4.15 Node Power Reduction

- Object: mocaNodePowerReduction
- Data Type: mocaPowerReduction

Encoding for the Data Type mocaNodePowerReduction:

OCTET STRING of per channel list of Transmit Power Control levels

NUM_VALUES	1 Byte	Function of mocaLinkType: Primary – 1 Bonded – 2 300 MHz MoCA 2.5 PHY – 3 400 MHz MoCA 2.5 PHY – 4 500 MHz MoCA 2.5 PHY – 5
{		
TPC	1 Byte	Transmit Power Control in dB used for this Link
} [NUM_VALUES]		

## 5 Appendix B

### 5.1 Algorithm to calculate MoCA Password Hash

MoCA Password Hash is a 16-bit value determined by performing SHA-1 on a 0-extended password concatenated with a SALT value. First, if the password is less than 64 printable ASCII characters, the password bytes are prepended with ASCII '0' in order to create a 64 character value. Second, the 64 character result is concatenated with the static SALT string 'MocaPasswordHash' to create the SHA input. The SHA-1 algorithm is performed on the SHA input to create a 160-bit output. The 16-LSBs of the SHA-1 output are extracted as the password hash.

```
uint16 CalcMoCAHash (void)
```

```
{
```

```
uint8 passWrd[MOCA_MAX_PASSWORD_LENGTH];
```

```
uint8 shaInput[MOCA_MAX_PASSWORD_LENGTH+CONCAT_STRING_LENGTH];
```

```
uint16 PWD_SHA1[10];
```

```
memset(passWrd, 0x30, sizeof(passWrd));
```

```
// pre-fill with 0x30
```

```
GetPassword ((void *)passWrd, MOCA_MAX_PASSWORD_LENGTH);
```

```
// stuff password into passWrd array
```

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)





## 6 MoCA Access 2.5-MIB Definition

```
--
-- *****
-- MOCA-ACCESS-25-MIB.my: MoCA Access 2.5 Device MIB.
--
-- Copyright (c) 2017-2018 by Multimedia over Coax Alliance
-- All rights reserved.
--
-- These MIBs must be used with the companion memo
-- "MoCA Enterprise Structure of Management Information",
-- MoCA-SMI-V1.3-20171122, November 22, 2017.
-- *****
--
```

```
MOCA-ACCESS-25-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY,
    OBJECT-TYPE,
    NOTIFICATION-TYPE,
    Integer32,
    Unsigned32,
    Counter32,
    Counter64
    FROM SNMPv2-SMI
    MODULE-COMPLIANCE,
    OBJECT-GROUP,
    NOTIFICATION-GROUP
    FROM SNMPv2-CONF
    TEXTUAL-CONVENTION,
    DisplayString,
    MacAddress,
    RowStatus,
    TruthValue
    FROM SNMPv2-TC
    ifIndex
    FROM IF-MIB
    mocaMIBs
    FROM MOCA-SMI;
```

```
mocaAccess25 MODULE-IDENTITY
```

```
    LAST-UPDATED "201805230000Z" -- May 23, 2018 00:00 AM
    ORGANIZATION "Multimedia over Coax Alliance (MoCA) Technical
        Working Group."
    CONTACT-INFO "www.mocalliance.org
```

```

    Postal:Multimedia over Coax Alliance
        2400 Camino Ramon
        Suite 375
        San Ramon, CA 94583
```

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

```

        USA
        Tel:      +1 925 275-6606
        Email:    help@mocalliance.org"
DESCRIPTION
    "Initial version of this MIB module. It defines the objects
    used to remote configure, monitor, and diagnose the
    Multimedia over Coax Alliance (MoCA) network. The following
    Specifications are referenced in this MIB definition,
    1. MoCA Access MAC/PHY Specification v2.5, Nov 22, 2017 .
    The key changes from the MoCA 2.0 MIB in this revision are,
    a. Initial MoCA Access MIB started from MoCA 2.0 MIB."
REVISION    "201805230000Z"
DESCRIPTION "Original"
    ::= { mocaMIBs 4 }

--
-- Textual Conventions
--

MocaVersion ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
        "Represent the MoCA Specification version.
        mocaAccess2dot5 represents MoCA Access Specification v2.5"

    SYNTAX     INTEGER {
                mocaAccess2dot5 (25)
            }

MocaNodeID ::= TEXTUAL-CONVENTION
    DISPLAY-HINT "d"
    STATUS     current
    DESCRIPTION
        "Represent a MoCA node ID. MoCA Access v2.5 network can have a maximum
        of 64 nodes, so node ID is 0 to 63."
    SYNTAX     Unsigned32 (0..63)

MocaNodeMask ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
        "Bit mask which identifies the Node IDs of a set of nodes"
    SYNTAX     BITS {
                mocaNodeID0 (0),
                mocaNodeID1 (1),
                mocaNodeID2 (2),
                mocaNodeID3 (3),
                mocaNodeID4 (4),
                mocaNodeID5 (5),
                mocaNodeID6 (6),
                mocaNodeID7 (7),
            }
```

mocaNodeID8 (8),  
mocaNodeID9 (9),  
mocaNodeID10 (10),  
mocaNodeID11 (11),  
mocaNodeID12 (12),  
mocaNodeID13 (13),  
mocaNodeID14 (14),  
mocaNodeID15 (15),  
mocaNodeID16 (16),  
mocaNodeID17 (17),  
mocaNodeID18 (18),  
mocaNodeID19 (19),  
mocaNodeID20 (20),  
mocaNodeID21 (21),  
mocaNodeID22 (22),  
mocaNodeID23 (23),  
mocaNodeID24 (24),  
mocaNodeID25 (25),  
mocaNodeID26 (26),  
mocaNodeID27 (27),  
mocaNodeID28 (28),  
mocaNodeID29 (29),  
mocaNodeID30 (30),  
mocaNodeID31 (31),  
mocaNodeID32 (32),  
mocaNodeID33 (33),  
mocaNodeID34 (34),  
mocaNodeID35 (35),  
mocaNodeID36 (36),  
mocaNodeID37 (37),  
mocaNodeID38 (38),  
mocaNodeID39 (39),  
mocaNodeID40 (40),  
mocaNodeID41 (41),  
mocaNodeID42 (42),  
mocaNodeID43 (43),  
mocaNodeID44 (44),  
mocaNodeID45 (45),  
mocaNodeID46 (46),  
mocaNodeID47 (47),  
mocaNodeID48 (48),  
mocaNodeID49 (49),  
mocaNodeID50 (50),  
mocaNodeID51 (51),  
mocaNodeID52 (52),  
mocaNodeID53 (53),  
mocaNodeID54 (54),  
mocaNodeID55 (55),  
mocaNodeID56 (56),  
mocaNodeID57 (57),

```

        mocaNodeID58 (58),
        mocaNodeID59 (59),
        mocaNodeID60 (60),
        mocaNodeID61 (61),
        mocaNodeID62 (62),
        mocaNodeID63 (63)
    }

```

MocaBand ::= TEXTUAL-CONVENTION

```

STATUS    current
DESCRIPTION
    "This bitmask represents the MoCA Bands and Subbands the device is
    configured to operate in. A '1' in a bit represents the band is
    supported"
SYNTAX    BITS {
        mocareserved0 (0),
        mocaBandAA    (1),
        mocaBandAB    (2),
        mocaBandAC    (3),
        mocaBandAD    (4),
        mocaBandAE    (5),
        mocareserved6 (6),
        mocareserved7 (7)
    }

```

MocaPrivacy ::= TEXTUAL-CONVENTION

```

STATUS    current
DESCRIPTION
    "Represents the MoCA Access PrivacySupported bitMask "
SYNTAX    BITS {
        mocaAccessPrivacy (0),
        reserved1         (1),
        reserved2         (2),
        reserved3         (3),
        reserved4         (4),
        reserved5         (5),
        reserved6         (6),
        reserved7         (7)
    }

```

-- Update to mocaFreq???MHz

MocaChannel ::= TEXTUAL-CONVENTION

```

STATUS    current
DESCRIPTION
    "Represent the RF center frequency of a MoCA channel in MHz.
    Note that enumerated value uses syntax mocaFreq<freqInMhz>MHz
    Band A-A Moca2.0 Bonded-PHY Channel Frequencies
    AA1      center frequency at 450 MHz
    AA2      center frequency at 475 MHz

```

A1 center frequency at 500 MHz  
A2 center frequency at 525 MHz  
A3 center frequency at 550 MHz  
A4 center frequency at 575 MHz  
A5 center frequency at 600 MHz  
AA3 center frequency at 625 MHz  
AA4 center frequency at 650 MHz  
A6 center frequency at 675 MHz  
A7 center frequency at 700 MHz  
A8 center frequency at 725 MHz  
A9 center frequency at 750 MHz  
A10 center frequency at 775 MHz  
A11 center frequency at 800 MHz  
A12 center frequency at 825 MHz  
A13 center frequency at 850 MHz  
Band A-A MoCA2.5 Phy Channel Frquencies (1st Channel)  
A2p5-1 center frequence at 450 MHz  
A2p5-2 center frequence at 475 MHz  
A2p5-3 center frequence at 500 MHz  
A2p5-4 center frequence at 525 MHz  
A2p5-5 center frequence at 550 MHz  
A2p5-6 center frequence at 575 MHz  
A2p5-7 center frequence at 600 MHz  
A2p5-8 center frequence at 625 MHz  
A2p5-9 center frequence at 650 MHz  
Band A-B Moca2.0 Bonded-PHY Channel Frequencies  
B1 center frequency at 825 MHz  
B2 center frequency at 850 MHz  
B3 center frequency at 875 MHz  
B4 center frequency at 900 MHz  
B5 center frequency at 925 MHz  
B6 center frequency at 950 MHz  
B7 center frequency at 975 MHz  
B8 center frequency at 1000 MHz  
B9 center frequency at 1024 MHz  
B10 center frequency at 1050 MHz  
B11 center frequency at 1075 MHz  
B12 center frequency at 1100 MHz  
BB1 center frequency at 1125 MHz  
B13 center frequency at 1150 MHz  
BB2 center frequency at 1175 MHz  
B14 center frequency at 1200 MHz  
BB3 center frequency at 1225 MHz  
B15 center frequency at 1250 MHz  
BB4 center frequency at 1275 MHz  
B16 center frequency at 1300 MHz  
BB5 center frequency at 1325 MHz  
B17 center frequency at 1350 MHz  
BB6 center frequency at 1375 MHz  
B18 center frequency at 1400 MHz

BB7 center frequency at 1425 MHz  
 B19 center frequency at 1450 MHz  
 BB8 center frequency at 1475 MHz  
 B20 center frequency at 1500 MHz  
 BB9 center frequency at 1525 MHz  
 B21 center frequency at 1550 MHz  
 BB10 center frequency at 1575 MHz  
 B22 center frequency at 1600 MHz  
 BB11 center frequency at 1625 MHz  
 Band A-B Moca2.5 Phy Channel Frequencies (1st Channel)  
 B2p5-1 center frequency at 850 MHz  
 B2p5-2 center frequency at 875 MHz  
 B2p5-3 center frequency at 900 MHz  
 B2p5-4 center frequency at 925 MHz  
 B2p5-5 center frequency at 950 MHz  
 B2p5-6 center frequency at 975 MHz  
 B2p5-7 center frequency at 1000 MHz  
 B2p5-8 center frequency at 1024 MHz  
 B2p5-9 center frequency at 1050 MHz  
 B2p5-10 center frequency at 1075 MHz  
 B2p5-11 center frequency at 1100 MHz  
 B2p5-12 center frequency at 1125 MHz  
 B2p5-13 center frequency at 1150 MHz  
 B2p5-14 center frequency at 1175 MHz  
 B2p5-15 center frequency at 1200 MHz  
 B2p5-16 center frequency at 1225 MHz  
 B2p5-17 center frequency at 1250 MHz  
 B2p5-18 center frequency at 1275 MHz  
 B2p5-19 center frequency at 1300 MHz  
 B2p5-20 center frequency at 1325 MHz  
 B2p5-21 center frequency at 1350 MHz  
 B2p5-22 center frequency at 1375 MHz  
 B2p5-23 center frequency at 1400 MHz  
 B2p5-24 center frequency at 1425 MHz  
 Band A-C Moca2.0 Bonded-PHY Channel Frequencies  
 CC1 center frequency at 1075 MHz  
 CC2 center frequency at 1100 MHz  
 CC3 center frequency at 1125 MHz  
 C1 center frequency at 1150 MHz  
 CC4 center frequency at 1175 MHz  
 C2 center frequency at 1200 MHz  
 CC5 center frequency at 1225 MHz  
 C3 center frequency at 1250 MHz  
 CC6 center frequency at 1275 MHz  
 C4 center frequency at 1300 MHz  
 CC7 center frequency at 1325 MHz  
 C5 center frequency at 1350 MHz  
 CC8 center frequency at 1375 MHz  
 C6 center frequency at 1400 MHz  
 CC9 center frequency at 1425 MHz

C7 center frequency at 1450 MHz  
CC10 center frequency at 1475 MHz  
C8 center frequency at 1500 MHz  
CC11 center frequency at 1525 MHz  
C9 center frequency at 1550 MHz  
CC12 center frequency at 1575 MHz  
C10 center frequency at 1600 MHz  
CC13 center frequency at 1625 MHz  
Band A-C Moca2.5 Phy Channel Frequencies (1st Channel)  
C2p5-1 center frequency at 1075 MHz  
C2p5-2 center frequency at 1100 MHz  
C2p5-3 center frequency at 1125 MHz  
C2p5-4 center frequency at 1150 MHz  
C2p5-5 center frequency at 1175 MHz  
C2p5-6 center frequency at 1200 MHz  
C2p5-7 center frequency at 1225 MHz  
C2p5-8 center frequency at 1250 MHz  
C2p5-9 center frequency at 1275 MHz  
C2p5-10 center frequency at 1300 MHz  
C2p5-11 center frequency at 1325 MHz  
C2p5-12 center frequency at 1350 MHz  
C2p5-13 center frequency at 1375 MHz  
C2p5-14 center frequency at 1400 MHz  
C2p5-15 center frequency at 1425 MHz  
Band A-D Moca2.0 Bonded-PHY Channel Frequencies  
D1 center frequency at 1150 MHz  
DD1 center frequency at 1175 MHz  
D2 center frequency at 1200 MHz  
DD2 center frequency at 1225 MHz  
D3 center frequency at 1250 MHz  
DD3 center frequency at 1275 MHz  
D4 center frequency at 1300 MHz  
DD4 center frequency at 1325 MHz  
D5 center frequency at 1350 MHz  
DD5 center frequency at 1375 MHz  
D6 center frequency at 1400 MHz  
DD6 center frequency at 1425 MHz  
D7 center frequency at 1450 MHz  
DD7 center frequency at 1475 MHz  
D8 center frequency at 1500 MHz  
DD8 center frequency at 1525 MHz  
D9 center frequency at 1550 MHz  
DD9 center frequency at 1575 MHz  
D10 center frequency at 1600 MHz  
DD10 center frequency at 1625 MHz  
Band A-D Moca2.5 PHY Channel Frequencies (1st Channel)  
D2p5-1 center frequency at 1175 MHz  
D2p5-2 center frequency at 1200 MHz  
D2p5-3 center frequency at 1225 MHz  
D2p5-4 center frequency at 1250 MHz



```

D2p5-5 center frequency at 1275 MHz
D2p5-6 center frequency at 1300 MHz
D2p5-7 center frequency at 1325 MHz
D2p5-8 center frequency at 1350 MHz
D2p5-9 center frequency at 1375 MHz
D2p5-10 center frequency at 1400 MHz
D2p5-11 center frequency at 1425 MHz
Band A-E Moca2.0 Bonded-PHY Channel Frequencies
E1 center frequency at 1400 MHz
EE1 center frequency at 1425 MHz
E2 center frequency at 1450 MHz
EE2 center frequency at 1475 MHz
E3 center frequency at 1500 MHz
EE3 center frequency at 1525 MHz
E4 center frequency at 1550 MHz
EE4 center frequency at 1575 MHz
E5 center frequency at 1600 MHz
EE5 center frequency at 1625 MHz
Band A-E Moca2.5 PHY Channel Frequencies (1st Channel)
E2p5-1 center frequency at 1425 MHz"
REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 15."
SYNTAX INTEGER {
    unknown (0),
    mocaFreq450MHz (450),
    mocaFreq475MHz (475),
    mocaFreq500MHz (500),
    mocaFreq525MHz (525),
    mocaFreq550MHz (550),
    mocaFreq575MHz (575),
    mocaFreq600MHz (600),
    mocaFreq625MHz (625),
    mocaFreq650MHz (650),
    mocaFreq675MHz (675),
    mocaFreq700MHz (700),
    mocaFreq725MHz (725),
    mocaFreq750MHz (750),
    mocaFreq775MHz (775),
    mocaFreq800MHz (800),
    mocaFreq825MHz (825),
    mocaFreq850MHz (850),
    mocaFreq875MHz (875),
    mocaFreq900MHz (900),
    mocaFreq925MHz (925),
    mocaFreq950MHz (950),
    mocaFreq975MHz (975),
    mocaFreq1000MHz (1000),
    mocaFreq1025MHz (1025),
    mocaFreq1050MHz (1050),
    mocaFreq1075MHz (1075),
    mocaFreq1100MHz (1100),

```

```

mocaFreq1125MHz      (1125),
mocaFreq1150MHz      (1150),
mocaFreq1175MHz      (1175),
mocaFreq1200MHz      (1200),
mocaFreq1225MHz      (1225),
mocaFreq1250MHz      (1250),
mocaFreq1275MHz      (1275),
mocaFreq1300MHz      (1300),
mocaFreq1325MHz      (1325),
mocaFreq1350MHz      (1350),
mocaFreq1375MHz      (1375),
mocaFreq1400MHz      (1400),
mocaFreq1425MHz      (1425),
mocaFreq1450MHz      (1450),
mocaFreq1475MHz      (1475),
mocaFreq1500MHz      (1500),
mocaFreq1525MHz      (1525),
mocaFreq1550MHz      (1550),
mocaFreq1575MHz      (1575),
mocaFreq1600MHz      (1600),
mocaFreq1625MHz      (1625)
}

```

MocaChannelMask ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Represent one or more RF center frequency of MoCA channels using a bitmask. Bit 63 is the most significant bit, and bit 0 is the least significant bit. Not all bits are valid MoCA channels.

```

bit 0 center frequency at 400 MHz
bit 1 center frequency at 425 MHz
bit 2 center frequency at 450 MHz
bit 3 center frequency at 475 MHz
bit 4 center frequency at 500 MHz
bit 5 center frequency at 525 MHz
bit 6 center frequency at 550 MHz
bit 7 center frequency at 575 MHz
bit 8 center frequency at 600 MHz
bit 9 center frequency at 625 MHz
bit 10 center frequency at 650 MHz
bit 11 center frequency at 675 MHz
bit 12 center frequency at 700 MHz
bit 13 center frequency at 725 MHz
bit 14 center frequency at 750 MHz
bit 15 center frequency at 775 MHz
bit 16 center frequency at 800 MHz
bit 17 center frequency at 825 MHz
bit 18 center frequency at 850 MHz

```

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

```
bit 19 center frequency at 875 MHz
bit 20 center frequency at 900 MHz
bit 21 center frequency at 925 MHz
bit 22 center frequency at 950 MHz
bit 23 center frequency at 975 MHz
bit 24 center frequency at 1000 MHz
bit 25 center frequency at 1025 MHz
bit 26 center frequency at 1050 MHz
bit 27 center frequency at 1075 MHz
bit 28 center frequency at 1100 MHz
bit 29 center frequency at 1125 MHz
bit 30 center frequency at 1150 MHz
bit 31 center frequency at 1175 MHz
bit 32 center frequency at 1200 MHz
bit 33 center frequency at 1225 MHz
bit 34 center frequency at 1250 MHz
bit 35 center frequency at 1275 MHz
bit 36 center frequency at 1300 MHz
bit 37 center frequency at 1325 MHz
bit 38 center frequency at 1350 MHz
bit 39 center frequency at 1375 MHz
bit 40 center frequency at 1400 MHz
bit 41 center frequency at 1425 MHz
bit 42 center frequency at 1450 MHz
bit 43 center frequency at 1475 MHz
bit 44 center frequency at 1500 MHz
bit 45 center frequency at 1525 MHz
bit 46 center frequency at 1550 MHz
bit 47 center frequency at 1575 MHz
bit 48 center frequency at 1600 MHz
bit 49 center frequency at 1625 MHz
bit 50 center frequency at 1650 MHz
bit 51 center frequency at 1675 MHz
bit 52 center frequency at 1700 MHz
bit 53 center frequency at 1725 MHz
bit 54 center frequency at 1750 MHz
bit 55 center frequency at 1775 MHz
bit 56 center frequency at 1800 MHz
bit 57 center frequency at 1825 MHz
bit 58 center frequency at 1850 MHz
bit 59 center frequency at 1875 MHz
bit 60 center frequency at 1900 MHz
bit 61 center frequency at 1925 MHz
bit 62 center frequency at 1950 MHz
bit 63 center frequency at 1975 MHz "
SYNTAX BITS {
    freq400MHz (0),
    freq425MHz (1),
    freq450MHz (2),
    freq475MHz (3),
```

freq500MHz (4),  
freq525MHz (5),  
freq550MHz (6),  
freq575MHz (7),  
freq600MHz (8),  
freq625MHz (9),  
freq650MHz (10),  
freq675MHz (11),  
freq700MHz (12),  
freq725MHz (13),  
freq750MHz (14),  
freq775MHz (15),  
freq800MHz (16),  
freq825MHz (17),  
freq850MHz (18),  
freq875MHz (19),  
freq900MHz (20),  
freq925MHz (21),  
freq950MHz (22),  
freq975MHz (23),  
freq1000MHz (24),  
freq1025MHz (25),  
freq1050MHz (26),  
freq1075MHz (27),  
freq1100MHz (28),  
freq1125MHz (29),  
freq1150MHz (30),  
freq1175MHz (31),  
freq1200MHz (32),  
freq1225MHz (33),  
freq1250MHz (34),  
freq1275MHz (35),  
freq1300MHz (36),  
freq1325MHz (37),  
freq1350MHz (38),  
freq1375MHz (39),  
freq1400MHz (40),  
freq1425MHz (41),  
freq1450MHz (42),  
freq1475MHz (43),  
freq1500MHz (44),  
freq1525MHz (45),  
freq1550MHz (46),  
freq1575MHz (47),  
freq1600MHz (48),  
freq1625MHz (49),  
freq1650MHz (50),  
freq1675MHz (51),  
freq1700MHz (52),  
freq1725MHz (53),

```

freq1750MHz (54),
freq1775MHz (55),
freq1800MHz (56),
freq1825MHz (57),
freq1850MHz (58),
freq1875MHz (59),
freq1900MHz (60),
freq1925MHz (61),
freq1950MHz (62),
freq1975MHz (63)
}

```

```

MocaPhyRate ::= TEXTUAL-CONVENTION
  DISPLAY-HINT "d"
  STATUS current
  DESCRIPTION
    "Represent the transmit PHY rate in Mbps."
  SYNTAX Unsigned32 (0..2048)

```

```

MocaDB ::= TEXTUAL-CONVENTION
  DISPLAY-HINT "d"
  STATUS current
  DESCRIPTION
    "Represent one decibel or 1 dB."
  SYNTAX Unsigned32 (0..127)

```

```

MocaDBm ::= TEXTUAL-CONVENTION
  DISPLAY-HINT "d"
  STATUS current
  DESCRIPTION
    "Represent a measure of power in mW expressed in decibels, and
    calculated as follows:
    power = 10*log10( Vrms^2 / R * 1000 )
    where Vrms is the root-mean-square Voltage of the received
    waveform and R is 75 ohms."
  SYNTAX Integer32 (-128..127)

```

```

MocaScMod ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Represent the sub-carrier modulation"
  REFERENCE "MoCA Access MAC/PHY Specification v2.5"
  SYNTAX INTEGER {
    unused (0),
    bsk (1),
    qpsk (2),
    qam8 (3),
    qam16 (4),
    qam32 (5),
    qam64 (6),

```

```
    qam128 (7),  
    qam256 (8),  
    qam512 (9),  
    qam1024 (10)  
}
```

```
MocaPerMode ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Represent the packet error rate mode in used in the node. The value
    used define the NPER and the VLPER"
  REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 16.3.1 and
    section 17, table 17.1."
  SYNTAX INTEGER {
    nper (0),
    vlper(1) }
```

```
MocaSupportedBands ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Represent all the supported bands and sub-bands by this device"
  REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 15"
  SYNTAX BITS {
    noBand (0),
    bandAA (1),
    bandAB (2),
    bandAC (3),
    bandAD (4),
    bandAE (5),
    reserved6 (6),
    reserved7 (7)
  }
```

```
MocaPowerState ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Represent the power state defined by MoCA2.0 specification the
    different power states are:
    Power state M0: Active
    Power state M1: Low Power Idle
    Power State M2: Standby
    Power State M3: Sleep"
  REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 11 "
  SYNTAX INTEGER {
    m0Active (0),
    m1LowPowerIdle (1),
    m2Standby (2),
    reserved3 (3)
  }
```

```

MocaPowerStateBits ::= TEXTUAL-CONVENTION
    STATUS    current
    DESCRIPTION
        "Represent the power state defined by MoCA2.0 specification the
        different power states are:
            Power state M0: Active
            Power state M1: Low Power Idle
            Power State M2: Standby
            Power State M3: Sleep"
    REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 11 "
    SYNTAX    BITS {
        m0Active          (0),
        m1LowPowerIdle    (1),
        m2Standby         (2),
        reserved3         (3),
        reserved4         (4),
        reserved5         (5),
        reserved6         (6),
        reserved7         (7)
    }

MocaAcaStatus ::= TEXTUAL-CONVENTION
    STATUS    current
    DESCRIPTION
        "Represent the status of the last Aca probe"
    SYNTAX    INTEGER {
        success           (0),
        failBADCHANNEL    (1),
        failNOEVMPROBE   (2),
        fail              (3),
        inPROGRESS        (4)
    }

MocaChannelIndex ::= TEXTUAL-CONVENTION
    STATUS    current
    DESCRIPTION
        "Represent the MoCA Access Channels primary, secondary, 1st, 2nd, 3rd,
        4th, and 5th channels."
    SYNTAX    INTEGER {
        primary          (0),
        secondary        (1),
        first            (2),
        second           (3),
        third            (4),
        fourth           (5),
        fifth            (6)
    }

```



```

mocaNotifications OBJECT IDENTIFIER ::= { mocaAccess25 0 }
mocaObjects        OBJECT IDENTIFIER ::= { mocaAccess25 1 }
mocaConformance   OBJECT IDENTIFIER ::= { mocaAccess25 2 }

--
-- MoCA Objects
--

mocaIfConfigTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaIfConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "MoCA interface configuration table. This table supports the
        configuration of RF frequency, transmit power, link privacy, and
        traps related parameters.

        This table is indexed by ifIndex.

        Entries in this table cannot be created or deleted by the
        network management system. All entries are created or deleted
        by the device software."
    ::= { mocaObjects 1 }

mocaIfConfigEntry OBJECT-TYPE
    SYNTAX      MocaIfConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in the MoCA interface configuration table. Each
        device can have one or more MoCA interfaces. Each interface has
        a unique MAC address.

        There is a one to one relationship between a MoCA interface and
        an interface defined in other IETF MIB. The same 'ifIndex' can
        be used to identify MoCA objects as well as other IETF MIB
        objects."
    INDEX      { ifIndex }
    ::= { mocaIfConfigTable 1 }

MocaIfConfigEntry ::= SEQUENCE {
    mocaIfAccessPassword      DisplayString,
    mocaIfAuthenticationExempt OCTET STRING,
    mocaIfBand                 MocaBand,
    mocaIfChannelMask          MocaChannelMask,
    mocaIfClientResetReqNodeMask MocaNodeMask,
    mocaIfClientResetReqSend   Unsigned32,
    mocaIfEnable               TruthValue,
    mocaIfFirstOffset          Integer32,

```

```

mocaIfLof                               MocaChannel,
mocaIfLofUpdateEnable                   TruthValue,
mocaIfNetworkNameAdmissionRules        INTEGER,
mocaIfNetworkNameNcNn                   DisplayString,
mocaIfNumChannels                        Unsigned32,
mocaIfPer25Mode                          MocaPerMode,
mocaIfPerCentPriority                     OCTET STRING,
mocaIfPhyThreshold                       MocaPhyRate,
mocaIfPowerControlTargetRate            MocaPhyRate,
mocaIfPowerStateRequest                  MocaPowerState,
mocaIfPrimaryOffset                      Integer32,
mocaIfPrivacyEnable                      TruthValue,
mocaIfPrivacyNetAatekRefresh            Unsigned32,
mocaIfPrivacyPairwAapmkRefresh          Unsigned32,
mocaIfPrivacyPairwAatekRefresh          Unsigned32,
mocaIfRlapmEnable                        TruthValue,
mocaIfRlapmProfileSelect                 Unsigned32,
mocaIfSapmEnable                         TruthValue,
mocaIfSapmProfileSelect                 Unsigned32,
mocaIfSecondaryOffset                    Integer32,
mocaIfTpcEnable                          TruthValue,
mocaIfTpcTargetRateNper                  MocaPhyRate,
mocaIfTrafficPermissionEthertype        Unsigned32,
mocaIfTrapClientResetFailEn            TruthValue,
mocaIfTrapClientResetSuccessEn         TruthValue,
mocaIfTrapConnectedNodesChangeEn       TruthValue,
mocaIfTrapLmoEn                         TruthValue,
mocaIfTrapNcPrivSupportedRecEn         TruthValue,
mocaIfTrapPhyThresholdEn                TruthValue,
mocaIfTrapPowerStateEn                  TruthValue,
mocaIfTrapStatusChangeEn                TruthValue,
mocaIfTxPowerLimit                       MocaDB,
mocaIfPowerContour                       INTEGER
}

```

mocaIfAccessPassword OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..64))

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Specify the MoCA password using an ASCII numeric string.

This value must be 12 to 64 ASCII printable characters with codes in the range 33 to 126 inclusive. Access is only allowed when used with SNMPv3 protocol, but not allowed when used with SNMPv1 or SNMPv2c protocol.

If this value changes, this node will drop from the network if mocaIfPrivacyEnable is true. If this managed object is changed, it may not be possible to change this object again via

the SNMP protocol. The mechanism to change this object again is out of the scope of this MIB definition draft."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 10.3.1"

DEFVAL { "" }

::= { mocaIfConfigEntry 1 }

mocaIfAuthenticationExempt OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (0..760 ))

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Packet Array of GUIDs (8 OCTETS) that indicates which Client Nodes are exempted from authentication. The parameter is used by the Access NC Only. Contains 0 to 63 elements or 4 to 508 OCTETS GUID of 64 1's indicates all GUID's are exempt.

```
struct mocaAuthenticationExempt {
    unsigned int NUM_ADDRS; // 4 octets (Number of GUIDs in the list)
    struct {
        char    GUID[8];    // 8 octets (Node GUID MAC Address)
    } [NUM_ADDRS];
};"
```

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.5"

::= { mocaIfConfigEntry 2 }

mocaIfBand OBJECT-TYPE

SYNTAX MocaBand

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Configure the band or sub-band for the Node to operate in."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 15."

::= { mocaIfConfigEntry 3 }

mocaIfChannelMask OBJECT-TYPE

SYNTAX MocaChannelMask

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Specify a list of RF center frequencies represented by a bitmask which this MoCA node is allowed to form or join a MoCA network. If the new list of frequencies does not contain the frequency this MoCA node is tuned to, this node must drop from the network.

If this managed object is changed, it may not be possible to change this object again via the SNMP protocol. The mechanism to change this object again is out of the scope of this MIB definition draft."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 7.1.4."

DEFVAL { { freq1150MHz,

```

        freq1200MHz,
        freq1250MHz,
        freq1300MHz,
        freq1350MHz,
        freq1400MHz,
        freq1450MHz,
        freq1500MHz,
        freq1550MHz,
        freq1600MHz } }
 ::= { mocaIfConfigEntry 4 }

mocaIfClientResetReqNodeMask OBJECT-TYPE
    SYNTAX      MocaNodeMask
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Initiates a ClientReset PIE with the RESET_NODE_BITMASK set to this
        object. A bit set to 1 in the ith bit position indicates that the
        Client Node will be reset."
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 7.13."
    DEFVAL     { { } }
    ::= { mocaIfConfigEntry 5 }

mocaIfClientResetReqSend OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Request the Access NC to send Client Reset PIEs with the
        RESET_NODE_BITMASK set to 'mocaIfClientResetReqNodeMask'."
    ::= { mocaIfConfigEntry 6 }

mocaIfEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Enable the MoCA interface if this value is 'true'. Disable the
        MoCA interface if this value is 'false'."

        If this managed object is disabled, it may not be possible to
        enable this object via the SNMP protocol. The mechanism to
        re-enable this object is out of the scope of this MIB definition
        draft."
    DEFVAL     { true }
    ::= { mocaIfConfigEntry 7 }

```

```

mocaIfFirstOffset OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the the offset of the First Channel's center frequency
        below the center frequency of the Primary Channel. Valid values are
        -4 to 0."
    ::= { mocaIfConfigEntry 8 }

mocaIfLof OBJECT-TYPE
    SYNTAX      MocaChannel
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Indicate the MoCA channel in MHz this interface is tuned to
        when it was last in the 'linkUp' state.

        If this interface is never part of a MoCA network, report the
        factory default Last Operational Frequency (LOF)."

```

```
 ::= { mocaIfConfigEntry 13 }
mocaIfNetworkNameNcNn OBJECT-TYPE
    SYNTAX      DisplayString (SIZE (0..32))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The Network Name the Node is to use for any MoCA network it creates,
        or indicating no name to use for the network.  When
        mocaIfNetworkNameAdmissionRules==(NAME) this parameter also indicates
        to the Node the name of the network to join."
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 8.3.2."
    DEFVAL     { "" }
    ::= { mocaIfConfigEntry 14 }

mocaIfNumChannels OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The maximum number of 100MHz channels the Node uses when the Node
        forms the network.  1-5"
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 8.3.2."
    DEFVAL     { 0 }
    ::= { mocaIfConfigEntry 15 }

mocaIfPer25Mode OBJECT-TYPE
    SYNTAX      MocaPerMode
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Controls which transmission PER mode the Node uses for MPDUs"
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 8.3.5.6
        and Table 17-1."
    DEFVAL     { nper }
    ::= { mocaIfConfigEntry 16 }
```

```
mocaIfPerCentPriority OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..40))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Packed Structure defining the maximum percent of wire time to
        allocate to all the priority levels
        struct mocaPerCentPriority {
            char    MAX_PRI;    // 1 octet  (Max Priority Level in enum [0..7])
            struct {
                char    UplinkPercentage; // 1 octet  (in percentage [0..100])
                char    DownlinkPercentage; // 1 octet  (in percentage [0..100])
            } [MAX_PRI];
        };"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.2.4."
    DEFVAL    { '00'H }
    ::= { mocaIfConfigEntry 17 }

mocaIfPhyThreshold OBJECT-TYPE
    SYNTAX      MocaPhyRate
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specify a minimum PHY rate in Mbps.

        If transmit PHY rate between all pairs of MoCA nodes are greater
        than or equal to this value, then transmit PHY rate of one pair
        of MoCA node is less than this value, send
        'mocaTrapBelowPhyThreshold' if 'mocaIfTrapPhyThresholdEn' is
        'true'.

        If transmit PHY rate between one or more pairs of MoCA nodes are
        less than this value, then transmit PHY rate between all pairs
        of MoCA nodes are greater than or equal to this value, send
        'mocaTrapAbovePhyThreshold' if 'mocaIfTrapPhyThresholdEn' is
        'true'."
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, Section 3, see
        Unusable Link definition."
    DEFVAL    { 123 }
    ::= { mocaIfConfigEntry 18 }
```

```
mocaIfPowerControlTargetRate OBJECT-TYPE
    SYNTAX      MocaPhyRate
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specify the target transmit PHY rate in Mbps for the power
        control algorithm. This change should be used in the next
        maintenance cycle in the MoCA network."
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 16.3.8."
    DEFVAL      { 630 }
    ::= { mocaIfConfigEntry 19 }

mocaIfPowerStateRequest OBJECT-TYPE
    SYNTAX      MocaPowerState
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Carries a request to the Node to transition into a specified Power
        State"
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 11."
    DEFVAL      { m0Active }
    ::= { mocaIfConfigEntry 20 }

mocaIfPrimaryOffset OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the the offset of the Primary Channel's center frequency
        relative to the center frequency of the Beacon Channel. Valid values
        are 0 (centered with beacon), 25 (above beacon), -25 (below beacon)
        in MHz"
    ::= { mocaIfConfigEntry 21 }

mocaIfPrivacyEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Enable link privacy if this value is true, and use the
        mocaIfPassword to generate the MAC management key and initial
        privacy management key. Disable link privacy and do not perform
        link encryption if this value is false.

        This node will drop from the network if this value changes.
```



If this managed object is disabled, it may not be possible to enable this object via the SNMP protocol. The mechanism to re-enable this object is out of the scope of this MIB definition draft."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 10."

DEFVAL { false }

::= { mocaIfConfigEntry 22 }

mocaIfPrivacyNetAatekRefresh OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Dynamic Network AATEK Refresh Period in Minutes. Max: 6 hours."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 10."

DEFVAL { 360 }

::= { mocaIfConfigEntry 23 }

mocaIfPrivacyPairwAapmkRefresh OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Dynamic Pairwise AAPMK Refresh Period in Minutes. Max: 72 hours."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 10."

DEFVAL { 4320 }

::= { mocaIfConfigEntry 24 }

mocaIfPrivacyPairwAatekRefresh OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Dynamic Pairwise AATEK Refresh Period in Minutes. Max: 6 hours."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 10."

DEFVAL { 360 }

::= { mocaIfConfigEntry 25 }

```
mocaIfRlapmEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Controls whether the RLAPM function is enabled and the Node needs to
        apply it, or disabled and the Node needs to ignore it.  RLAPM
        function enabled
        when set to true RLAPM function disabled when set to false
        Default value is false"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.12 and
        Table 17-1."
    DEFVAL     { false }
    ::= { mocaIfConfigEntry 27 }

mocaIfRlapmProfileSelect OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Controls which RLAPM profile is active there is up to 61 profile"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.12.2 and
        Table 17-1."
    ::= { mocaIfConfigEntry 28 }

mocaIfSapmEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Controls whether the SAPM function is enabled and the Node needs to
        apply it, or disabled and the Node needs to ignore it When set to
        true the SAPM function is enabled When set to false the SAPM function
        is disabled
        Default value is disabled (false)"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.12.1 and
        Table 17-1."
    DEFVAL     { false }
    ::= { mocaIfConfigEntry 29 }

mocaIfSapmProfileSelect OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Controls which SAPM profile is active"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.12.1 and
        Table 17-1."
    ::= { mocaIfConfigEntry 30 }
```

```
mocaIfSecondaryOffset OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the the offset of the Secondary Channel's center frequency
        relative to the center frequency of the Primary Channel. Valid
        values are 125 (above Primary), -125 (below Primary) in MHz"
    ::= { mocaIfConfigEntry 31 }

mocaIfTpcEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Enable automatic power control if this value is true. Use a fixed
        transmit power level if this value is false."
    DEFVAL     { true }
    ::= { mocaIfConfigEntry 32 }

mocaIfTpcTargetRateNper OBJECT-TYPE
    SYNTAX      MocaPhyRate
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The Target PHY Rate of the receiving MoCA 2.0 Node in order to
        optimize the long-term PHY rate and transmit power level between the
        transmitting and receiving MoCA nodes when TPC is enabled and
        PERMODE = NPER"
    ::= { mocaIfConfigEntry 33 }

mocaIfTrafficPermissionEthertype OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This controls the EtherType value of MSDUs that are allowed to be
        transmitted/received to/from a Non-Secured Admitted Client Node."
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, Section 10.7."
    DEFVAL     { '888E'H }
    ::= { mocaIfConfigEntry 34 }
```

```
mocaIfTrapClientResetFailEn OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Enables 'mocaTrapClientResetFail'
         - When set to true it allows the MoCA node to report a trap when
           Client Reset Fails
         - When set to false no trap is reported when Client Reset Fails
         - Default value is false"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.13."
    DEFVAL    { false }
    ::= { mocaIfConfigEntry 35 }

mocaIfTrapClientResetSuccessEn OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Enables 'mocaTrapClientResetSuccess'
         - When set to true it allows the MoCA node to report a trap when
           Client Reset Succeeds
         - When set to false no trap is reported when Client Reset Succeeds
         - Default value is false"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.13."
    DEFVAL    { false }
    ::= { mocaIfConfigEntry 36 }

mocaIfTrapConnectedNodesChangeEn OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "If 'mocaIfTrapConnectedNodesChangeEn' is 'true' and 'mocaIfNumNodes'
         changes, then the trap 'mocaTrapConnectedNodesChange' is sent.

         Only one node in the MoCA network should set 'mocaIfNumNodes'
         to 'true'."
    DEFVAL    { false }
    ::= { mocaIfConfigEntry 37 }
```

## mocaIfTrapLmoEn OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Used for LMO message trap

- When set to true it allows the MoCA node to report a trap on transitioning to LMO
- When set to false no trap report the MoCA node is not transitioning to LMO
- Default value is false"

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 18."

DEFVAL { false }

::= { mocaIfConfigEntry 38 }

## mocaIfTrapNcPrivSupportedRecEn OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Enables 'mocaTrapNcPrivacySupportedRec'

- When set to true it allows the MoCA node to report a trap when Permanent Salt IE received
- When set to false no trap is reported when Permanent Salt IE is received
- Default value is false"

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 8.3.2.2."

DEFVAL { false }

::= { mocaIfConfigEntry 39 }

## mocaIfTrapPhyThresholdEn OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"If transmit PHY rate between all pairs of MoCA nodes are greater than or equal to mocaIfPhyThreshold, then transmit PHY rate of one pair of MoCA node is less than mocaIfPhyThreshold, send mocaTrapBelowPhyThreshold if this value is true, do not send mocaTrapBelowPhyThreshold if this value is false.

If transmit PHY rate between one or more pairs of MoCA nodes are less than mocaIfPhyThreshold, then transmit PHY rate between all pairs of MoCA nodes are greater than or equal to mocaIfPhyThreshold, send mocaTrapAbovePhyThreshold if this value is true, do not send mocaTrapAbovePhyThreshold if this value is false.

Only one node in the MoCA network should set

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).

All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

```
'mocaIfPhyThresholdEn' to 'true'."
DEFVAL { false }
 ::= { mocaIfConfigEntry 40 }

mocaIfTrapPowerStateEn OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "Used for power state message trap
    - When Set to true it allows the MoCA node to report a trap on
      each power state transition
      - mocaTrapPowerStateBcstRec
      - mocaTrapPowerStateMONc
      - mocaTrapPowerStateResp
      - mocaTrapPowerStateTrnsReq
      - mocaTrapPowerStateUcstPen
      - mocaTrapPowerStateWupUr
    - When set to false no trap report is generated when the node
      power state changes
    - Dealut value is false"
REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 11."
DEFVAL { false }
 ::= { mocaIfConfigEntry 41 }

mocaIfTrapStatusChangeEn OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "If 'mocaIfTrapStatusChangeEn' is 'true' and 'mocaIfStatus'
    changes, then the trap 'mocaTrapIfStatusChange' is sent."
DEFVAL { false }
 ::= { mocaIfConfigEntry 42 }

mocaIfTxPowerLimit OBJECT-TYPE
SYNTAX MocaDB
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "Specify the transmit power BACKOFF in dB. The transmit power
    SHALL only be changed when the device mocaIfStatus object is
    not linkUp, which means the interface is not part of a MoCA
    network."
REFERENCE "MoCA MAC/PHY Specification v2.0 , Section 7.11."
DEFVAL { 0 }
 ::= { mocaIfConfigEntry 43 }

mocaIfPowerContour OBJECT-TYPE
SYNTAX INTEGER {
```

```

        pc1 (1),
        pc2 (2),
        pc3 (3)
    }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "Power Contour used by the Access NC and reported in the Power Contour
    Network IE within the Discovery Response"
REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.3.2."
 ::= { mocaIfConfigEntry 44 }

mocaIfRlapmTable OBJECT-TYPE
SYNTAX SEQUENCE OF MocaIfRlapmEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Sequence of Rlapm entry"
 ::= { mocaObjects 2 }

mocaIfRlapmEntry OBJECT-TYPE
SYNTAX MocaIfRlapmEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Rlapm entry table"
INDEX {
    ifIndex,
    mocaIfRlapmProfile,
    mocaIfRlapmFrequency
}
 ::= { mocaIfRlapmTable 1 }

MocaIfRlapmEntry ::= SEQUENCE {
    mocaIfRlapmProfile Unsigned32,
    mocaIfRlapmFrequency Unsigned32,
    mocaIfRlapmGarpl Unsigned32,
    mocaIfRlapmPhyMargin Unsigned32,
    mocaIfRlapmStatus RowStatus
}

mocaIfRlapmProfile OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "RLAPM profile index used for the RLAPM table."
 ::= { mocaIfRlapmEntry 1 }

```

```
mocaIfRlapmFrequency OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "MoCA frequency (band) index used for the RLAPM table."
    ::= { mocaIfRlapmEntry 2 }

mocaIfRlapmGarpl OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        " Global Aggregated Receive Power Level (GARPL). GARPLi = -Ni dBm
        where Ni = Unsigned32 in the range of 0 to 65 and 'i' spans
        61 values."
    ::= { mocaIfRlapmEntry 3 }

mocaIfRlapmPhyMargin OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Receive Level Added Phy Margin(RLAPM). RLAPMi = 0.5ni dB where
        'ni' = Unsigned32 in the range of 0 to 60 and 'i' spans 61 values."
    ::= { mocaIfRlapmEntry 4 }

mocaIfRlapmStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "True if all of the following are 'true': 'mocaIfRlapmEnable' is
        enabled, 'mocaIfRlapmProfileSelect' is equal to 'mocaIfRlapmProfile',
        and 'mocaIfRlapmFrequency' corresponds to either the primary or
        secondary if profile C link, or one of first through fifth channels
        if profile D link."
    ::= { mocaIfRlapmEntry 5 }

mocaIfSapmTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaIfSapmEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Sequence of Sapm entry"
    ::= { mocaObjects 3 }
```



```

mocaIfSapmEntry OBJECT-TYPE
    SYNTAX      MocaIfSapmEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Sapm table entry"
    INDEX       {
        ifIndex,
        mocaIfSapmProfile,
        mocaIfSapmFrequency
        }
    ::= { mocaIfSapmTable 1 }

```

```

MocaIfSapmEntry ::= SEQUENCE {
    mocaIfSapmProfile      Unsigned32,
    mocaIfSapmFrequency   Unsigned32,
    mocaIfSapmARPLTHLD    Unsigned32,
    mocaIfSapmPhyMargin   OCTET STRING,
    mocaIfSapmStatus      RowStatus
}

```

```

mocaIfSapmProfile OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "profile index used for the SAPM table."
    ::= { mocaIfSapmEntry 1 }

```

```

mocaIfSapmFrequency OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "frequency (band) index used for the SAPM table."
    ::= { mocaIfSapmEntry 2 }

```

```

mocaIfSapmARPLTHLD OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An Aggregate Received Power Level Threshold (ARPL_THLD) from 0 to 65
        (representing dBm) in steps of 1 dB."
    ::= { mocaIfSapmEntry 3 }

```

```

mocaIfSapmPhyMargin OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Packet array of SAPM values (in dB) for each OFDM subcarrier in the
         range 0 dB to 60.0 dB in 0.5 dB steps.  For each available subcarrier
         'ni', a value of 0.5ni dB where 'ni' = Unsigned32 in the range of
         0 to 120."
    ::= { mocaIfSapmEntry 4 }

mocaIfSapmStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "True if all of the following are 'true': 'mocaIfSapmEnable' is
         enabled, 'mocaIfSapmProfileSelect' is equal to 'mocaIfSapmProfile',
         and 'mocaIfSapmFrequency' corresponds to either the primary or
         secondary if profile C link, or one of first through fifth channels
         if profile D link."
    ::= { mocaIfSapmEntry 5 }

mocaIfAcaTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaIfAcaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The Interface mocaIfAcaTable,
         is used by the managed node to initiate On Demand LMO.
         The managed node is identified by the interface index (ifIndex)."
    ::= { mocaObjects 4 }

mocaIfAcaEntry OBJECT-TYPE
    SYNTAX      MocaIfAcaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in the MoCA Aca table."
    INDEX      {
        ifIndex
        }
    ::= { mocaIfAcaTable 1 }

MocaIfAcaEntry ::= SEQUENCE {
    mocaIfAcaChannel      Unsigned32,
    mocaIfAcaInitiate     TruthValue,
    mocaIfAcaNodeID       MocaNodeID,
    mocaIfAcaPowerProfile OCTET STRING,

```

```

    mocaIfAcaReportNodeMask      MocaNodeMask,
    mocaIfAcaStatus              MocaAcaStatus,
    mocaIfAcaStatusTrapCompleted TruthValue,
    mocaIfAcaTotalRxPower        MocaDBm,
    mocaIfAcaType                 INTEGER
}

mocaIfAcaChannel OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The channel number under assessment, starting from 0 in increments of
        25MHz"
    ::= { mocaIfAcaEntry 1 }

mocaIfAcaInitiate OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "An ACA request is initiated when this parameter is written"
    ::= { mocaIfAcaEntry 2 }

mocaIfAcaNodeID OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The Source Node id."
    ::= { mocaIfAcaEntry 3 }

mocaIfAcaPowerProfile OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Binary string array with 1 byte for each subcarrier"
    ::= { mocaIfAcaEntry 4 }

mocaIfAcaReportNodeMask OBJECT-TYPE
    SYNTAX      MocaNodeMask
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the Nodes that are requested to be part of the channel
        assessment:
        Setting bits corresponding to Node IDs of these Nodes to 1
        (LSB corresponds to Node ID 0x0)"
    ::= { mocaIfAcaEntry 5 }

```

```
mocaIfAcaStatus OBJECT-TYPE
    SYNTAX      MocaAcaStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Status of the previous ACA probe"
    ::= { mocaIfAcaEntry 6 }

mocaIfAcaStatusTrapCompleted OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicated when Power Profile is ready"
    ::= { mocaIfAcaEntry 7 }

mocaIfAcaTotalRxPower OBJECT-TYPE
    SYNTAX      MocaDBm
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "dBm (-128 to +127 dBm)"
    ::= { mocaIfAcaEntry 8 }

mocaIfAcaType OBJECT-TYPE
    SYNTAX      INTEGER {
        quiet (0),
        evm   (1)
    }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The ACA type is either 'EVM' or 'QUIET'"
    ::= { mocaIfAcaEntry 9 }

mocaIfStatusTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaIfStatusEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "MoCA interface status information table. This table provides
        features supported and operation parameters of the MoCA
        interface.

        This table is indexed by ifIndex.

        Entries in this table cannot be created or deleted by the
        network management system. All entries are created or deleted by
```

```

    the device software."
 ::= { mocaObjects 5 }

```

```

mocaIfStatusEntry OBJECT-TYPE
    SYNTAX      MocaIfStatusEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in the MoCA interface status information table."
    INDEX       { ifIndex }
    ::= { mocaIfStatusTable 1 }

```

```

MocaIfStatusEntry ::= SEQUENCE {
    mocaIfAeNumber          Unsigned32,
    mocaIfAggregationSize  Unsigned32,
    mocaIfAuthenticationResult  OCTET STRING,
    mocaIfBwMgmtLinkParameters  OCTET STRING,
    mocaIfBwMgmtNetParameters  OCTET STRING,
    mocaIfBwMgmtNodeParameters  OCTET STRING,
    mocaIfChannelSupport      MocaChannelMask,
    mocaIfClientResetFailNodeMask  MocaNodeMask,
    mocaIfConnectedNodesInfo  OCTET STRING,
    mocaIfFlowInfo           OCTET STRING,
    mocaIfLinkDownCount      Counter32,
    mocaIfLinkUpTime         Unsigned32,
    mocaIfLmoNodeID         MocaNodeID,
    mocaIfMacAddress         MacAddress,
    mocaIfMaxIngressNodeBw  MocaPhyRate,
    mocaIfMaxEgressNodeBw  MocaPhyRate,
    mocaIfMocaVersion       MocaVersion,
    mocaIfName              DisplayString,
    mocaIfNcPrivacySupported  MocaPrivacy,
    mocaIfNetworkNamePayload  DisplayString,
    mocaIfNetworkState      INTEGER,
    mocaIfNetworkVersion    MocaVersion,
    mocaIfNodeID            MocaNodeID,
    mocaIfNumNodes          Unsigned32,
    mocaIfNumPriority        Unsigned32,
    mocaIfPasswordHash      DisplayString,
    mocaIfPduNumber         Unsigned32,
    mocaIfPfsPrivacyProvision  MocaNodeMask,
    mocaIfPowerStateCap     MocaPowerStateBits,
    mocaIfResetCount        Counter32,
    mocaIfResetReason       DisplayString,
    mocaIfRFChannel         MocaChannel,
    mocaIfSoftwareVersion   OCTET STRING,
    mocaIfStatus            INTEGER,
    mocaIfSupportedBands    MocaSupportedBands,
    mocaIfTxGcdRate         MocaPhyRate,
    mocaIfPrivacySupported  MocaPrivacy,

```

```

mocaIfLinkState          OCTET STRING,
mocaIfBeaconTxPwr       Integer32,
mocaIfBeaconRxPwr       Integer32,
mocaIfNcPowerContour    INTEGER,
mocaIfNcVersion          Unsigned32
}

mocaIfAeNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Maximum number of allocation elements, excluding the TAUs and the
        Dummy DAUs, in one MAP the node can process."
    ::= { mocaIfStatusEntry 1 }

mocaIfAggregationSize OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the maximum number of bytes in one aggregate which can
        be received by the node."
    ::= { mocaIfStatusEntry 2 }

mocaIfAuthenticationResult OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (8))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Packed structure representing the result from Authentication Process.
        struct mocaAuthenticationResult {
            long    ClientNodeIdVector; // 8 octets
                                     // Bit value in the ith bit positions
                                     // indicates NodeID=(i) is: (LSB=bit0)
                                     // successfully authenticated (1),
                                     // unauthenticated (0),
                                     // Bit 0 (Access NC Node ID)
                                     // is always set
        };"
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 6.3.5"
    ::= { mocaIfStatusEntry 3 }

mocaIfBwMgmtLinkParameters OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (12..524))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Packed structure array representing the parameter
        BW_MGMTlink_parameters for up to 63 nodes

```

```

struct mocaBwMgmtmLinkParameters {
    long    ClientNodeIdVector;    // 8 octets
                                        // Bit value in the ith bit positions
                                        // indicates NodeID=(i) (LSB=bit0)
    int     NUM_NODES;            // 4 octets (Number of Node IDs
                                        // indicated in the
                                        // ClientNodeIdVector)

    struct {
        short UplinkPHYRate;        // 2 octets (in Mbps)
        short UpLinkBurstOverhead;  // 2 octets (in uS)
        short DownlinkPHYRate;      // 2 octets (in Mbps)
        short DownLinkBurstOverhead; // 2 octets (in uS)
    } [NUM_NODES];
};"

```

REFERENCE "MoCA Access MAC/PHY Specification v2.5, Secions 24 and 17, Table 17-1."

::= { mocaIfStatusEntry 4 }

mocaIfBwMgmtNetParameters OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (3))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Packed structure representing the parameter BW\_MGMTnet\_parameters

```

struct mocaBwMgmtNetParameters {
    short MinRRInterval;    // 2 octets (in uS)
    char  NetworkOverhead;  // 1 octet (in percentage [0..100])
};"

```

REFERENCE "MoCA Access MAC/PHY Specification v2.5, Secions 24 and 17, Table 17-1."

::= { mocaIfStatusEntry 5 }

mocaIfBwMgmtNodeParameters OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (12..524))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Packed structure representing the parameter BW\_MGMTnode\_parameters

```

struct mocaBwMgmtNodeParameters {
    long    NodeIdVector;    // 8 octets
                                        // Bit value in the ith bit positions
                                        // indicates NodeID=(i) (LSB=bit0)
    int     NUM_NODES;      // 4 octets (Number of Node IDs
                                        // indicated in the NodeIdVector)

    struct {
        int    MaxBytesAggregation; // 4 octets (in bytes)
        int    MaxPacketsAggregation; // 4 octets (in number of packets)
    } [NUM_NODES];
};"

```

REFERENCE "MoCA Access MAC/PHY Specification v2.5, Secions 24 and 17,

```

        Table 17-1."
 ::= { mocaIfStatusEntry 6 }

mocaIfChannelSupport OBJECT-TYPE
    SYNTAX      MocaChannelMask
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specify a list of RF center frequencies represented by a
        bitmask which this MoCA node can support to form or join a MoCA
        network."
 ::= { mocaIfStatusEntry 7 }

mocaIfClientResetFailNodeMask OBJECT-TYPE
    SYNTAX      MocaNodeMask
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Reports the Node IDs of Client Nodes that failed to ACK the
        ClientReset PIE. A bit set to 1 in the ith bit position indicates
        that the Client Node with Node ID = i failed to ACK."
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, section 7.13."
    DEFVAL      { { } }
 ::= { mocaIfStatusEntry 8 }

mocaIfConnectedNodesInfo OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (12..588))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Packed structure array representing information for each node
        connected in the network for up to 64 nodes
        struct mocaConnectedNodesInfo {
            long    NodeIdVector;           // 8 octets
                                           // Bit value in the ith bit positions
                                           // indicates NodeID=(i) (LSB=bit0)
            int     NUM_NODES;             // 4 octets (Number of Node IDs
                                           // indicated in the NodeIdVector)

            struct {
                char    GUID[8];          // 8 octets (Node GUID MAC Address)
                char    MocaVersion;     // 1 octet (MoCA Version)
            } [NUM_NODES];
        };"
    REFERENCE   "MoCA Access MAC/PHY Specification v2.5, Secions 17,
        Table 17-1."
 ::= { mocaIfStatusEntry 9 }

```



```
mocaIfFlowInfo OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Packet Array of MAC Addresses (6 OCTETs) that reports all the
        Multicast Flows and Unicast Flows.
        struct mocaFlowInfo {
            int      NUM_ADDRS;          // 4 octets (Number of addrs in the list)
            struct {
                char  MacAddress[6]; // 6 octets (MAC Address)
            } [NUM_ADDRS];
        };"

    REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.5"
    ::= { mocaIfStatusEntry 10 }

mocaIfLinkDownCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the number of times that this node has lost link with the
        MoCA network."
    ::= { mocaIfStatusEntry 11 }

mocaIfLinkUpTime OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the total number of seconds that this interface is
        part of a MoCA network.

        This may be used with 'sysUpTime' to determine the link
        availability in the MoCA interface. Note that 'sysUpTime' is in
        10 milliseconds and 'mocaIfLinkUpTime' is in seconds."
    ::= { mocaIfStatusEntry 12 }

mocaIfLmoNodeID OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the ID of the node currently undergoing LMO."
    ::= { mocaIfStatusEntry 13 }
```

```
mocaIfMacAddress OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the MAC address of this MoCA interface. This MAC
        address is encoded in the first six bytes of the Globally Unique
        Identifier (GUID).

        For example, MoCA interface with MAC address aa:bb:cc:dd:ee:ff
        will have a GUID of aa:bb:cc:dd:ee:ff:00:00."
    ::= { mocaIfStatusEntry 14 }

mocaIfMaxIngressNodeBw OBJECT-TYPE
    SYNTAX      MocaPhyRate
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the maximum bandwidth of this MoCA interface for
        admission of flows if this node is an ingress node."
    REFERENCE  "MoCA MAC/PHY Specification v1.1, section 3.17.2.2.2."
    ::= { mocaIfStatusEntry 15 }

mocaIfMaxEgressNodeBw OBJECT-TYPE
    SYNTAX      MocaPhyRate
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the maximum bandwidth of this MoCA interface for
        admission of Flows if this node is an egress node."
    REFERENCE  "MoCA MAC/PHY Specification v1.1, section 3.17.2.2.2."
    ::= { mocaIfStatusEntry 16 }

mocaIfMocaVersion OBJECT-TYPE
    SYNTAX      MocaVersion
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the MoCA version supported by this MoCA interface reported in
        the NODE_PROTOCOL_SUPPORT field of the AdmissionRequest."
    ::= { mocaIfStatusEntry 18 }

mocaIfName OBJECT-TYPE
    SYNTAX      DisplayString (SIZE (0..16))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the textual name of this MoCA interface. The value of
        this object should be the name of the interface as assigned by
        the MoCA device.
```

Since MoCA has an Ethernet convergence layer, this name may be 'ethN' where 'N' is the port number."  
 ::= { mocaIfStatusEntry 19 }

mocaIfNcPrivacySupported OBJECT-TYPE

SYNTAX MocaPrivacy

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The latest NC\_PRIVACY\_SUPPORTED value received from the Access NC. Before any NC\_PRIVACY\_SUPPORTED is received, reflects the value of the parameter PRIVACYsupported.

Bit0: MoCA Access Privacy Not Supported (0)

MoCA Access Privacy Supported (1)"

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 8.3.2."

DEFVAL { { } }

::= { mocaIfStatusEntry 20 }

mocaIfNetworkNamePayload OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..32))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The payload of the latest received Network Name from the Access NC."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 8.3.2."

DEFVAL { " " }

::= { mocaIfStatusEntry 21 }

mocaIfNetworkState OBJECT-TYPE

SYNTAX INTEGER {

beginNodeAdmissionState (1),

newNodeTypeOneProbeTxState (2),

newNodeTypeOneProbeRxState (3),

beginPhyProfileState (5),

steadyState (6),

typeThreeProbeState (7),

lmoTypeOneProbeState (8),

lmoNodeGcdDistributionState (9),

beginLmoPhyProfileState (10),

lmoGcdTypeOneProbeLinkState (11),

alternateChannelQuietLineState (12),

alternateChannelEvmProbeState (13),

unsolicitedProbeReportState (14),

beginUnsolicitedPhyProfileState (15),

rxDeterminedProbeState (16)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).

All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

"Specifies the current MoCA network state of the node."  
 REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.4.1."  
 ::= { mocaIfStatusEntry 22 }

mocaIfNetworkVersion OBJECT-TYPE

SYNTAX MocaVersion

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the MoCA version used in this MoCA network. If this interface is not part of a MoCA network, report 'mocaIfMocaVersion'.

This value uses the MOCA\_VERSION in the Beacon Frame."

REFERENCE "MoCA Access MAC/PHY Access Specification v2.5, section 6.3.2."

::= { mocaIfStatusEntry 23 }

mocaIfNodeID OBJECT-TYPE

SYNTAX MocaNodeID

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the node ID of this MoCA interface. If this interface is not part of a MoCA network, report a value of '0'."

::= { mocaIfStatusEntry 24 }

mocaIfNumNodes OBJECT-TYPE

SYNTAX Unsigned32 (0..64)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This value corresponds to the number of nodes that this node communicates to in the MoCA network.

This value may be smaller than the number of nodes reported by the NC node."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 3.6.3.2."

::= { mocaIfStatusEntry 25 }

mocaIfNumPriority OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Reports the maximum number of priority levels that the Access NC can support. (4 or 8)"

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 12.1."

DEFVAL { 0 }

::= { mocaIfStatusEntry 26 }

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).

All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

```
mocaIfPasswordHash OBJECT-TYPE
    SYNTAX      DisplayString (SIZE (4))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specify the MoCA password HASH using an ASCII numeric
        String. The Hash is the two LSBs of a SHA1.
        This value must be 4 hexadecimal digits long. Access
        is optional when used with SNMP v1 of SNMP v2c protocol."
    ::= { mocaIfStatusEntry 27 }

mocaIfPduNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Represent the maximum number of Ethernet packets aggregated in
        a MoCA frame that can be received by this MoCA interface."
    ::= { mocaIfStatusEntry 28 }

mocaIfPfsPrivacyProvision OBJECT-TYPE
    SYNTAX      MocaNodeMask
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "A list of Node IDs that reports which Client Nodes have Perfect
        Forward Secrecy (PFS) with the Access NC."
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 7.2.4."
    DEFVAL     { { } }
    ::= { mocaIfStatusEntry 29 }

mocaIfPowerStateCap OBJECT-TYPE
    SYNTAX      MocaPowerStateBits
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Reports the Power State capabilities of the Node, i.e. which power
        states it can transition to."
    REFERENCE  "MoCA Access MAC/PHY Specification v2.5, section 11."
    DEFVAL     { { m0Active } }
    ::= { mocaIfStatusEntry 30 }

mocaIfResetCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The reset counter for this node."
    ::= { mocaIfStatusEntry 31 }
```

```
mocaIfResetReason OBJECT-TYPE
    SYNTAX    DisplayString (SIZE (0..80))
    MAX-ACCESS read-only
    STATUS    current
    DESCRIPTION
        "Describes the reason for a MoCA link reset or failure to join a
        network."
    ::= { mocaIfStatusEntry 32 }

mocaIfRFChannel OBJECT-TYPE
    SYNTAX    MocaChannel
    MAX-ACCESS read-only
    STATUS    current
    DESCRIPTION
        "Indicate the MoCA channel in MHz this interface is tuned to
        when part of a MoCA network. When not part of a MoCA network
        this value may not reflect the actual tuned channel. If
        'mocaIfEnable' is 'false', report a value of 'unknown'."
    ::= { mocaIfStatusEntry 33 }

mocaIfSoftwareVersion OBJECT-TYPE
    SYNTAX    OCTET STRING (SIZE (2..82))
    MAX-ACCESS read-only
    STATUS    current
    DESCRIPTION
        "Indicate the software version of the MoCA device and other Vendor
        Specific information. This should be the same as the product software
        version in the MoCA certificate.
        struct mocaSoftwareVersion {
            short  VendorId;          // 2 octets (MoCA Vendor ID)
            char   Information[80]; // 0-80 bytes of Vendor Specific Information
        };"
    ::= { mocaIfStatusEntry 34 }

mocaIfStatus OBJECT-TYPE
    SYNTAX    INTEGER {
        disabled (1),
        noLink   (2),
        linkUp   (3) }
    MAX-ACCESS read-only
    STATUS    current
    DESCRIPTION
        "Indicate the current status of the MoCA interface."
```

'disabled' indicates interface is disabled. It will not be possible to read the 'disabled' status of the MoCA interface if this managed node does not have another local port (e.g. Ethernet) where SNMP objects can be accessed.

'noLink' indicates interface is enabled but not part of a network.

'linkUp' indicates interface is enabled and in a network."

```
::= { mocaIfStatusEntry 35 }
```

mocaIfSupportedBands OBJECT-TYPE

SYNTAX MocaSupportedBands

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Lists all the supported bands and sub-bands by this device. MoCA reserve the assignment of the first 8 bits positions (left to right) as follows:

Bit 0: No Band

Bit 1: Band A-A

Bit 2: Band A-B

Bit 3: Band A-C

Bit 4: Band A-D

Bit 5: Band A-E

Bit 6: position is reserved

Bit 7: position is reserved"

```
::= { mocaIfStatusEntry 36 }
```

mocaIfTxGcdRate OBJECT-TYPE

SYNTAX MocaPhyRate

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the PHY rate in Mbps for the transmit traffic broadcast from this node. In a MoCA 2.0 or mixed mode network, this object reports the PHY rate of the profile (NPER or VLPER) determined by the PERmode parameter."

```
::= { mocaIfStatusEntry 37 }
```

mocaIfPrivacySupported OBJECT-TYPE

SYNTAX MocaPrivacy

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Privacy modes that are supported when mocaIfPrivacyEnable set to true.

Bit0: MoCA Access Privacy Not Supported (0)

MoCA Access Privacy Supported (1)"

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 8.3.2."

```
::= { mocaIfStatusEntry 38 }
```

```

mocaIfLinkState OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicates the link state of the MoCA Access Network at the time
        of a Link Failure.
        struct mocaLinkState {
            char LINK_STATE;      // LINK_STATE_II from the
                                // transmitted/received MAP at the time of
                                // the Link Failure
            char ACF_TYPE;       // ACF_TYPE from the
                                // transmitted/received BEACON at the time of
                                // the Link Failure
        };"
    REFERENCE "MoCA Access MAC/PHY Specification v2.5."
    ::= { mocaIfStatusEntry 39 }

mocaIfBeaconTxPwr OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Transmit Power in dBm of the Beacon as reported in the
        BEACON_TRANSMIT_POWER field of the beacon."
    REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.2."
    ::= { mocaIfStatusEntry 40 }

mocaIfBeaconRxPwr OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Receive Power in dBm of the Beacon at the Client Node. Attenuation in
        dB can be calculated between a Client Node and the Access NC
        using 'mocaIfBeaconTxPwr' - 'mocaIfBeaconRxPwr'"
    REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.2."
    ::= { mocaIfStatusEntry 41 }

mocaIfNcPowerContour OBJECT-TYPE
    SYNTAX      INTEGER {
        pc1 (1),
        pc2 (2),
        pc3 (3)
    }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Power Contour reported by the Access NC in the Power Contour Network IE

```



within the Discovery Response"  
 REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.3.2."  
 ::= { mocaIfStatusEntry 42 }

mocaIfNcVersion OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"MoCA Version of the Access NC as reported in the NODE\_PROTOCOL\_SUPPORT field of the Admission Response."

REFERENCE "MoCA Access MAC/PHY Specification v2.5, section 6.3.5."

::= { mocaIfStatusEntry 43 }

mocaIfStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF MocaIfStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"MoCA interface statistics table. This table provides statistics of the MoCA interface. This table is indexed by 'ifIndex'. Entries in this table cannot be created or deleted by the network management system. All entries are created or deleted by the device software."

::= { mocaObjects 6 }

mocaIfStatsEntry OBJECT-TYPE

SYNTAX MocaIfStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry in the MoCA interface statistics table. The packet counters do not wrap in less than an hour with only 32 bits. Therefore 32 instead of 64 bits counters are used except for byte counters."

INDEX {  
     ifIndex  
 }

::= { mocaIfStatsTable 1 }

MocaIfStatsEntry ::= SEQUENCE {

mocaIfRxBroadcast	Counter64,
mocaIfRxBytes	Counter64,
mocaIfRxCorrectedErrors	Counter64,
mocaIfRxDrops	Counter32,
mocaIfRxMulticast	Counter64,
mocaIfRxPackets	Counter64,
mocaIfRxUnicast	Counter64,
mocaIfTxBroadcast	Counter64,
mocaIfTxBytes	Counter64,
mocaIfTxDrops	Counter32,
mocaIfTxMulticast	Counter64,

```
mocaIfTxPackets          Counter64,
mocaIfTxUnicast         Counter64
}

mocaIfRxBroadcast OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of Broadcast Ethernet packets received by this
        MoCA interface."
    ::= { mocaIfStatsEntry 1 }

mocaIfRxBytes OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of bytes received by this MoCA interface in
        Ethernet packets."
    ::= { mocaIfStatsEntry 2 }

mocaIfRxCorrectedErrors OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of received Ethernet packets that have errors and
        are corrected by this MoCA interface."
    ::= { mocaIfStatsEntry 3 }

mocaIfRxDrops OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of scheduled MoCA frames that are not detected or
        have uncorrectable errors and are dropped by this MoCA interface."
    ::= { mocaIfStatsEntry 4 }

mocaIfRxMulticast OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of Multicast Ethernet packets received by this
        MoCA interface."
    ::= { mocaIfStatsEntry 5 }

mocaIfRxPackets OBJECT-TYPE
```

```
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Indicate the number of good Ethernet packets received by this
  MoCA interface."
 ::= { mocaIfStatsEntry 6 }

mocaIfRxUnicast OBJECT-TYPE
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Indicate the number of Unicast Ethernet packets received by this MoCA
  interface."
 ::= { mocaIfStatsEntry 7 }

mocaIfTxBroadcast OBJECT-TYPE
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Indicate the number of Broadcast Ethernet packets transmitted by this
  MoCA interface."
 ::= { mocaIfStatsEntry 8 }

mocaIfTxBytes OBJECT-TYPE
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Indicate the number of bytes transmitted by this MoCA interface in
  Ethernet Packets."
 ::= { mocaIfStatsEntry 9 }

mocaIfTxDrops OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Indicate the number of Ethernet packet requests made at the MAC_SAP and
  not transmitted by this MoCA interface, including drops due to required
  throttling as required by the specification (see REFERENCE)."
```

REFERENCE "MoCA Access MAC/PHY Specification v2.5."

```
 ::= { mocaIfStatsEntry 10 }

mocaIfTxMulticast OBJECT-TYPE
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
```

## DESCRIPTION

"Indicate the number of Multicast Ethernet packets transmitted by this MoCA interface."

```
::= { mocaIfStatsEntry 11 }
```

## mocaIfTxPackets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"Indicate the number of Ethernet packets transmitted by this MoCA interface."

```
::= { mocaIfStatsEntry 12 }
```

## mocaIfTxUnicast OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"Indicate the number of Unicast packets transmitted by this MoCA interface."

```
::= { mocaIfStatsEntry 13 }
```

## mocaLinkStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF MocaLinkStatsEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"MoCA Link Statistics table. This table contains an entry for every node in the MoCA network, excluding this MoCA interface.

This table is indexed by 'ifIndex' and 'mocaLinkNodeIndex'.

Entries in this table cannot be created or deleted by the network management system. All entries are created or deleted by the device software."

```
::= { mocaObjects 7 }
```

## mocaLinkStatsEntry OBJECT-TYPE

SYNTAX MocaLinkStatsEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"An entry in the MoCA link statistics table."

```
INDEX {
    ifIndex,
    mocaLinkNodeIndex
}
```

```
::= { mocaLinkStatsTable 1 }
```

```

MocaLinkStatsEntry ::= SEQUENCE {
    mocaLinkNodeIndex      MocaNodeID,
    mocaLinkTxDrops        Counter64,
    mocaLinkTxPackets      Counter64
}

mocaLinkNodeIndex OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The node ID is used as an index to the MoCA node Link Statistics
        table."
    ::= { mocaLinkStatsEntry 1 }

mocaLinkTxDrops OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of Unicast Ethernet packet transmission requests
        made at the MAC_SAP, destined to the associated node, and not
        transmitted by this MoCA interface."
    ::= { mocaLinkStatsEntry 2 }

mocaLinkTxPackets OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the number of Unicast Ethernet packets transmitted by
        this MoCA interface to the associated node."
    ::= { mocaLinkStatsEntry 3 }

mocaNodeTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaNodeEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "MoCA node status information table. This table contains an
        entry for every node in the MoCA network, excluding this MoCA
        interface.

        This table is indexed by 'ifIndex' and 'mocaNodeIndex'.

        Entries in this table cannot be created or deleted by the
        network management system. All entries are created or deleted by
        the device software."
    ::= { mocaObjects 8 }

```

```

mocaNodeEntry OBJECT-TYPE
    SYNTAX      MocaNodeEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in the MoCA node status information table."
    INDEX       {
                ifIndex,
                mocaNodeIndex
                }
    ::= { mocaNodeTable 1 }

MocaNodeEntry ::= SEQUENCE {
    mocaNodeIndex      MocaNodeID,
    mocaNodeAeNumber   Unsigned32,
    mocaNodeAggregationSize Unsigned32,
    mocaNodeMacAddress  MacAddress,
    mocaNodeMocaVersion MocaVersion,
    mocaNodePduNumber  Unsigned32,
    mocaNodePowerState MocaPowerState,
    mocaNodeRxCorrected Counter64,
    mocaNodeRxDrops    Counter32,
    mocaNodeRxPackets  Counter64,
    mocaNodePDelay     Integer32
}

mocaNodeIndex OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The node ID is used as an index to order the MoCA node status
        information table."
    ::= { mocaNodeEntry 1 }

mocaNodeAeNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Represent the maximum number of allocation elements, excluding the
        TAUs and the Dummy DAUs, in one MAP that this MoCA node supports."
    ::= { mocaNodeEntry 2 }

mocaNodeAggregationSize OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION

```

"Represent the maximum number of bytes that can be aggregated in a MoCA frame that this MoCA node supports."  
 ::= { mocaNodeEntry 3 }

mocaNodeMacAddress OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the MAC address of the MoCA node identified by 'mocaNodeIndex'. This MAC address is encoded in the first six bytes of the Globally Unique Identifier (GUID).

For example, MoCA node with MAC address aa:bb:cc:dd:ee:ff will have a GUID of aa:bb:cc:dd:ee:ff:00:00."

::= { mocaNodeEntry 4 }

mocaNodeMocaVersion OBJECT-TYPE

SYNTAX MocaVersion

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the MoCA version supported by the MoCA node identified by 'mocaNodeIndex' as reported in the NODE\_PROTOCOL\_SUPPORT field of the AdmissionResponse."

::= { mocaNodeEntry 5 }

mocaNodePduNumber OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Represent the maximum number of Ethernet packets aggregated in a MoCA frame that this MoCA node supports."

::= { mocaNodeEntry 6 }

mocaNodePowerState OBJECT-TYPE

SYNTAX MocaPowerState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Power State of the Node identify by the 'mocaNodeIndex'"

::= { mocaNodeEntry 7 }

mocaNodeRxCorrected OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the number of Ethernet packets received by this MoCA

```
interface that have errors and are corrected from the MoCA node
identified by 'mocaNodeIndex'."
 ::= { mocaNodeEntry 8 }
```

mocaNodeRxDrops OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the number of scheduled MoCA frames that are not detected or have uncorrectable errors and are dropped by this MoCA interface from the MoCA node identified by 'mocaNodeIndex'."

```
 ::= { mocaNodeEntry 9 }
```

mocaNodeRxPackets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicate the number of good Ethernet packets received by this MoCA interface from the MoCA node identified by 'mocaNodeIndex'."

```
 ::= { mocaNodeEntry 10 }
```

mocaNodePDelay OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Propagation delay measured in units of 10ns between the local Node and Node identified by 'mocaNodeIndex'."

```
 ::= { mocaNodeEntry 14 }
```

mocaP2mpTable OBJECT-TYPE

SYNTAX SEQUENCE OF MocaP2mpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"MoCA Point to Multi-point PHY rate table. This table contains the unicast transmit/receive PHY rates between then Access NC and all CPE nodes in the Network. First entry is the GCD rate from the Access NC.

This table is indexed by 'ifIndex'."

```
 ::= { mocaObjects 9 }
```

mocaP2mpEntry OBJECT-TYPE

SYNTAX MocaP2mpEntry

MAX-ACCESS not-accessible

STATUS current



```

DESCRIPTION
  "An entry in the MoCA point to multipoint PHY rate table."
INDEX      {
    ifIndex
  }
 ::= { mocaP2mpTable 1 }

MocaP2mpEntry ::= SEQUENCE {
    mocaP2mpPhyRates      OCTET STRING
}

mocaP2mpPhyRates OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (9..265))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicate the transmit PHY rate in Mbps from the Access NC to the nodes
        identified in the NodeIdVector.  Index zero used to represent the GCD
        phy Rate from the Access NC to the client nodes. Set rxPhyRate to zero
        for index zero.  (No GCD RX)
        struct mocaP2mpPhyRates {
            long  NodeIdVector; // 8 octet  (Bitmask of Nodes included)
            char  numNodes;     // 1 octets (Number of Nodes to Follow
            {
                short txPhyRate; // 2 octects (Tx or GCD phy rate in Mbps)
                short rxPhyRate; // 2 octects (Rx phy rate in Mbps)
            } [numNodes];
        };"
    ::= { mocaP2mpEntry 1 }

mocaScModTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MocaScModEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "MoCA Point to multi-point Sub-Carrier Modulation table.
        This table contains the Unicast transmit sub-carrier modulation
        between all pair of nodes in the MoCA Network. This table is indexed
        by 'ifIndex', 'mocaTxNodeId', and 'mocaRxNodeId'.

        Since 'mocaScModList' is the sub-carrier modulation from
        'mocaTxNodeId' to 'mocaRxNodeId', this table does
        not contain any entries with 'mocaTxNodeId' equals
        'mocaRxNodeId'. In addition, a MoCA network can have
        less than 64 nodes, hence some values of 'mocaTxNodeId'
        and 'mocaRxNodeId' in the range of 0 to 63 may not exist.
        'mocaLinkType' is used to indicate the format of 'mocaScModList' and
        can
        be Primary, Bonded, 300MHz MoCA 2.5 PHY, 400MHz MoCA 2.5 PHY, or 500MHz
        MoCA 2.5 PHY."

```

```

    Entries in this table cannot be created or deleted by the
    network management system. All entries are created or deleted by
    the device software."
 ::= { mocaObjects 10 }

mocaScModEntry OBJECT-TYPE
    SYNTAX      MocaScModEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in the MoCA Point to Multi-point sub-carrier modulation
        table."
    INDEX       {
        ifIndex,
        mocaTxNodeId,
        mocaRxNodeId
    }
 ::= { mocaScModTable 1 }

MocaScModEntry ::= SEQUENCE {
    mocaTxNodeId      MocaNodeID,
    mocaRxNodeId      MocaNodeID,
    mocaLinkType      INTEGER,
    mocaScModList     OCTET STRING,
    mocaNodePower     OCTET STRING,
    mocaNodeSNR       OCTET STRING,
    mocaNodePowerReduction  OCTET STRING}

mocaTxNodeId OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The node ID of the transmit MoCA node is used as one of the
        index to order the point to multipoint PHY rate table."
 ::= { mocaScModEntry 1 }

mocaRxNodeId OBJECT-TYPE
    SYNTAX      MocaNodeID
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The node ID of the receive MoCA node is used as one of the
        index to order the point to multipoint sub-carrier modulation."
 ::= { mocaScModEntry 2 }

mocaLinkType OBJECT-TYPE
    SYNTAX      INTEGER {
        primary      (0),

```

```

        bonded      (1),
        m25300MHz  (2),
        m25400MHz  (3),
        m25500MHz  (4)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Indicates type type of link which exists from the node with
    NodeId='mocaTxNodeId' to the node with NodeId='mocaRxNodeId'.
    The link types are:
        - Primary          (`mocaScModList' 512 Octects),
        - Bonded           ('mocaScModList' 1024 Octects),
        - 300 MHz MoCA 2.5 PHY ('mocaScModList' 1536 Octects),
        - 400 MHz MoCA 2.5 PHY ('mocaScModList' 2048 Octects),
        - 500 MHz MoCA 2.5 PHY ('mocaScModList' 2560 Octects)
    Each channel has its own sub-carrier modulation data."
 ::= { mocaScModEntry 3 }

```

## mocaScModList OBJECT-TYPE

```

SYNTAX OCTET STRING (SIZE(512..2560))
MAX-ACCESS read-only
STATUS current
DESCRIPTION

```

"Indicate the sub-carrier modulation from the MoCA node identified by 'mocaTxNodeId' to the MoCA node identified by 'mocaRxNodeId' with size based on mocaLinkType.

Sub-carriers are encoded in sequential bytes starting from the most significant byte to the lowest significant byte. The lowest significant byte carriers the sub-carriers modulation for the lowest frequency sub-carrier and the most significant byte carriers the sub-carrier modulation for the highest frequency sub-carrier.

If 'mocaTxNodeId' == 0 and 'mocaRxNodeId' == 0, the sub-carrier modulation encoded in this array is for the GCD channel from the Access NC to the CPE nodes.

For Primary Channel, Bonded Channel, 300 MHz MoCA 2.5 PHY, 400 MHz MoCA 2.5 PHY, and 500 MHz MoCA 2.5 PHY channels the number of bytes in this object is 512, 1024, 1536, 2048, 2560 respectively."

```
 ::= { mocaScModEntry 4 }
```

## mocaNodePower OBJECT-TYPE

```

SYNTAX OCTET STRING (SIZE(2..6))
MAX-ACCESS read-only
STATUS current
DESCRIPTION

```

"Per channel list of Transmit or Receive power levels in dBm. If

'mocaTxNodeId' is equal to 'mocaIfNodeID', then indicates the unicast transmit power in dBm to the 'mocaRxNodeId', otherwise indicates the unicast receive power in dBm from the 'mocaTxNodeId' For transmit power, the power should be max power for the channel less the Transmit Power Control. For received power, the power measurement should cover the preamble and/or at least a symbol time. The averaging method employed for this measurement is implementation specific and not specified. It is recommended but not required to utilize the LMO EVM probes for this measurement.

The field numValues is a function of 'mocaLinkType':

```

Primary          - 1
Bonded           - 2
300 MHz MoCA 2.5 PHY - 3
400 MHz MoCA 2.5 PHY - 4
500 MHz MoCA 2.5 PHY - 5

```

```

struct mocaPower {
  char  numValues;      // 1 octet  (Number of Values to follow)
  {
    char  power;        // 1 octet  (Tx or Rx Power Level in dB)
  } [numValues];
};"

```

REFERENCE "MoCA Access MAC/PHY Specification v2.5."

::= { mocaScModEntry 5 }

#### mocaNodeSNR OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(2..6))

MAX-ACCESS read-only

STATUS current

#### DESCRIPTION

"Per channel list of Signal to Noise Ratio (SNR) in dB. If 'mocaTxNodeId' is equal to 'mocaIfNodeID', then this object is not present, otherwise indicates the SNR based on the EVM probe from the node identified by 'mocaTxNodeId'

The field numValues is a function of 'mocaLinkType':

```

Primary          - 1
Bonded           - 2
300 MHz MoCA 2.5 PHY - 3
400 MHz MoCA 2.5 PHY - 4
500 MHz MoCA 2.5 PHY - 5

```

```

struct mocaSNR {
  char  numValues;      // 1 octet  (Number of Values to follow)
  {
    char  snr;          // 1 octet  (Rx SNR in dB)
  } [numValues];
};"

```

REFERENCE "MoCA Access MAC/PHY Specification v2.5."

::= { mocaScModEntry 6 }

#### mocaNodePowerReduction OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(2..6))

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).

All rights reserved. MoCA®.

[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

```

MAX-ACCESS read-only
STATUS current
DESCRIPTION
  "Per channel list of The Transmit Power Control (TPC) back-off in dB. If
  'mocaTxNodeId' is equal to 'mocaIfNodeID', then indicates the TPC in dB
  utilized to back-off transmission to the 'mocaRxNodeId', otherwise
  indicates the TPC in dB that the 'mocaTxNodeId' utilized to back-off
  its transmission. 'mocaNodePowerReduction' is identical to the TPC
  back-off utilized for transmission from the TPC backoff parameter
  TX_POWER_ADJUSTMENT_100MHZ.
  The field numValues is a function of 'mocaLinkType':
    Primary - 1
    Bonded - 2
    300 MHz MoCA 2.5 PHY - 3
    400 MHz MoCA 2.5 PHY - 4
    500 MHz MoCA 2.5 PHY - 5
  struct mocaPowerReduction {
    char numValues; // 1 octet (Number of Values to follow)
    {
      char tpc; // 1 octet (TPC in dB)
    } [numValues];
  };
REFERENCE "MoCA Access MAC/PHY Specification v2.5."
 ::= { mocaScModEntry 7 }

```

```

--
-- MoCA Notifications
--

```

```

mocaTrapAbovePhyThreshold NOTIFICATION-TYPE
OBJECTS { mocaP2mpPhyRates }
STATUS current
DESCRIPTION
  "If transmit PHY rate between one or more pairs of MoCA nodes
  are less than 'mocaIfPhyThreshold', then the transmit PHY rate
  between all pairs of MoCA nodes goes above or is equal to
  'mocaIfPhyThreshold', send this notification if
  'mocaIfTrapPhyThresholdEn' is 'true'."
 ::= { mocaNotifications 1 }

```

```

mocaTrapBelowPhyThreshold NOTIFICATION-TYPE
OBJECTS { mocaP2mpPhyRates }
STATUS current
DESCRIPTION
  "If transmit PHY rate between all pairs of MoCA nodes are
  greater than or equal to 'mocaIfPhyThreshold', then the transmit PHY
  rate of one pair of MoCA nodes drops below 'mocaIfPhyThreshold',
  send this notification if 'mocaIfTrapPhyThresholdEn' is 'true'.

  This notification sends the lowest PHY rate in the MoCA network,

```

of 'mocaP2mpPhyRates', which is the transmit PHY rate from the node specified with 'mocaTxNodeId', to the node specified with 'mocaRxNodeId'.

If multiple links have the same lowest PHY rate, sends the PHY rate for the link with the lowest 'mocaTxNodeId'. If multiple links with the same 'mocaTxNodeId' have the same lowest PHY rate, sends the PHY rate for the link with the lowest 'mocaRxNodeId'.

Network management system should access 'mocaP2mpTable' to find if there are additional links below the threshold bandwidth.

For example, MoCA 2.0 interface with 'ifIndex' 1 detects transmit PHY rate from 'mocaTxNodeId' 0 to 'mocaRxNodeId' 4 is 300 Mbps, and this link is the first link to drop below the 'mocaIfPhyThreshold', then this trap contains one variable binding of

```
mocaP2mpPhyRates.1.0.4 = 300"
 ::= { mocaNotifications 2 }
```

mocaTrapClientResetFail NOTIFICATION-TYPE

OBJECTS { ifIndex }

STATUS current

DESCRIPTION

"This trap is sent to the Management Entity to report that at least one Client Node set in 'mocaIfClientResetReqNodeMask' failed to ACK the Client Reset PIE if 'mocaIfTrapClientResetFailEn' is 'true'. Failing node(s) are available in 'mocaIfClientResetFailNodeMask'."

```
 ::= { mocaNotifications 3 }
```

mocaTrapClientResetSuccess NOTIFICATION-TYPE

OBJECTS { ifIndex }

STATUS current

DESCRIPTION

"This trap is sent to the Management Entity by the Access NC to report that all Clients specified in the 'mocaIfClientResetReqNodeMask' have successfully ACKed the Client Reset PIE if 'mocaIfTrapClientResetSuccessEn' is 'true'."

```
 ::= { mocaNotifications 4 }
```

mocaTrapConnectedNodesChange NOTIFICATION-TYPE

OBJECTS { ifIndex }

STATUS current

DESCRIPTION

"This trap is sent to the Management Entity to the report that a new node

has joined the network or an EN has dropped from the network if 'mocaIfTrapConnectedNodesChangeEn' is 'true'. 'mocaIfConnectedNodesInfo will be updated."

```
 ::= { mocaNotifications 5 }

mocaTrapIfStatusChange NOTIFICATION-TYPE
OBJECTS { mocaIfStatus }
STATUS current
DESCRIPTION
  "This trap is sent when 'mocaIfTrapStatusChangeEn' is 'true'
  and mocaIfStatus changes value."
 ::= { mocaNotifications 6 }

mocaTrapLmoStatus NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
  "When 'mocaIfTrapLmoEn' is 'true', reports that the Node is
  starting LMO"
 ::= { mocaNotifications 7 }

mocaTrapNcPrivacySupportedRec NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
  "This trap is sent to the Management Entity when the node has received
  NC_PRIVACY_SUPPORTED value which is available through the object
  'mocaIfNcPrivacySupported' if 'mocaIfTrapNcPrivSupportedRecEn' is
  'true'."
 ::= { mocaNotifications 8 }

mocaTrapPowerStateBcstRec NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
  "When the Node is in Power State M1 or M2, reports that the Node has
  received a Broadcast data MSDU, which is available at the data
  interface if 'mocaIfTrapPowerStateEn' is 'true'."
 ::= { mocaNotifications 9 }

mocaTrapPowerStateM0Nc NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
  "When a Node is in Power State M1 reports that it is going to move to
  Power State M0 due to the NCs instruction if 'mocaIfTrapPowerStateEn'
  is 'true'."
 ::= { mocaNotifications 10 }

mocaTrapPowerStateResp NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
```

```

    "Carries the response of the Node to a request (through POWER_STATEREQ)
    to transition into a desired Power State if 'mocaIfTrapPowerStateEn'
    is 'true'."
 ::= { mocaNotifications 11 }

mocaTrapPowerStateTrnsReq NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
    "When a Node is in Power State M1 or M2, requests to transition to
    Power State M0 if 'mocaIfTrapPowerStateEn' is 'true'"
 ::= { mocaNotifications 12 }

mocaTrapPowerStateUcstPen NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
    "When the Node is in Power State M2, reports that a Unicast data MSDU
    destined to the Node is pending if 'mocaIfTrapPowerStateEn'
    is 'true'."
 ::= { mocaNotifications 13 }

mocaTrapPowerStateWupUr NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
    "When the Node is in Power State M2, reports that a wakeup request
    from NC due to unspecified reasons if 'mocaIfTrapPowerStateEn'
    is 'true'."
 ::= { mocaNotifications 14 }

mocaTrapLinkFailure NOTIFICATION-TYPE
OBJECTS { ifIndex }
STATUS current
DESCRIPTION
    "Indicates a link failure. If the failure was an Admission or LMO
    failure, 'mocaIfLinkState' indicates in which link state the link was
    dropped."
 ::= { mocaNotifications 15 }

---
--- MoCA Conformance
---

mocaMIBCompliances OBJECT IDENTIFIER ::= { mocaConformance 1 }
mocaMIBGroups OBJECT IDENTIFIER ::= { mocaConformance 2 }

---
--- Compliance Statements
---
```



```

mocaMIBCompliance MODULE-COMPLIANCE
  STATUS      current
  DESCRIPTION
    "The compliance statement for the mandatory and
    optional groups."
  MODULE
    MANDATORY-GROUPS {
      mocaIfConfigGroup,
      mocaIfRlapmGroup,
      mocaIfSapmGroup,
      mocaIfAcaGroup,
      mocaIfStatusGroup,
      mocaIfStatsGroup,
      mocaLinkStatsGroup,
      mocaNodeGroup,
      mocaP2mpGroup,
      mocaLinkModGroup,
      mocaTrapGroup
    }

  GROUP      mocaIfStatusOptionalGroup
  DESCRIPTION
    "Implementation of Nc Privacy Supported,
    Privacy Supported, and Password Hash are optional."

  OBJECT     mocaIfNcPrivacySupported
  DESCRIPTION
    "Read access not required."

  OBJECT     mocaIfPrivacySupported
  DESCRIPTION
    "Read access not required."

  OBJECT     mocaIfPasswordHash
  DESCRIPTION
    "Read access not required."

  GROUP      mocaTrapOptionalGroup
  DESCRIPTION
    "Implementation of network status change and LMO status are
    optional"

 ::= { mocaMIBCompliances 1 }

---
--- Units of Conformance
---
mocaIfConfigGroup OBJECT-GROUP
  OBJECTS    {

```

```

    mocaIfAccessPassword,
    mocaIfAuthenticationExempt,
    mocaIfBand,
    mocaIfChannelMask,
    mocaIfClientResetReqNodeMask,
    mocaIfClientResetReqSend,
    mocaIfEnable,
    mocaIfFirstOffset,
    mocaIfLof,
    mocaIfLofUpdateEnable,
    mocaIfNetworkNameAdmissionRules,
    mocaIfNetworkNameNcNn,
    mocaIfNumChannels,
    mocaIfPer25Mode,
    mocaIfPerCentPriority,
    mocaIfPhyThreshold,
    mocaIfPowerControlTargetRate,
    mocaIfPowerStateRequest,
    mocaIfPrimaryOffset,
    mocaIfPrivacyEnable,
    mocaIfPrivacyNetAatekRefresh,
    mocaIfPrivacyPairwAapmkRefresh,
    mocaIfPrivacyPairwAatekRefresh,
    mocaIfRlapmEnable,
    mocaIfRlapmProfileSelect,
    mocaIfSapmEnable,
    mocaIfSapmProfileSelect,
    mocaIfSecondaryOffset,
    mocaIfTpcEnable,
    mocaIfTpcTargetRateNper,
    mocaIfTrafficPermissionEthertype,
    mocaIfTrapClientResetFailEn,
    mocaIfTrapClientResetSuccessEn,
    mocaIfTrapConnectedNodesChangeEn,
    mocaIfTrapLmoEn,
    mocaIfTrapNcPrivSupportedRecEn,
    mocaIfTrapPhyThresholdEn,
    mocaIfTrapPowerStateEn,
    mocaIfTrapStatusChangeEn,
    mocaIfTxPowerLimit,
    mocaIfPowerContour
}
STATUS    current
DESCRIPTION
    "Parameters to configure the MoCA interface."
 ::= { mocaMIBGroups 1 }

mocaIfRlapmGroup OBJECT-GROUP
    OBJECTS {
        mocaIfRlapmGarpl,

```

```
        mocaIfRlapmPhyMargin,
        mocaIfRlapmStatus          }
STATUS    current
DESCRIPTION
    "Parameters to configure RLPAM for the MoCA interface."
 ::= { mocaMIBGroups 2 }

mocaIfSapmGroup OBJECT-GROUP
OBJECTS    {
    mocaIfSapmARPLTHLD,
    mocaIfSapmPhyMargin,
    mocaIfSapmStatus          }
STATUS    current
DESCRIPTION
    "Parameters to configure SAPM for the MoCA interface."
 ::= { mocaMIBGroups 3 }

mocaIfAcaGroup OBJECT-GROUP
OBJECTS    {
    mocaIfAcaChannel,
    mocaIfAcaInitiate,
    mocaIfAcaNodeID,
    mocaIfAcaPowerProfile,
    mocaIfAcaReportNodeMask,
    mocaIfAcaStatus,
    mocaIfAcaStatusTrapCompleted,
    mocaIfAcaTotalRxPower,
    mocaIfAcaType
    }
STATUS    current
DESCRIPTION
    "Parameters to control ACA for the MoCA interface."
 ::= { mocaMIBGroups 4 }

mocaIfStatusGroup OBJECT-GROUP
OBJECTS    {
    mocaIfAeNumber,
    mocaIfAggregationSize,
    mocaIfAuthenticationResult,
    mocaIfBwMgmtLinkParameters,
    mocaIfBwMgmtNetParameters,
    mocaIfBwMgmtNodeParameters,
    mocaIfChannelSupport,
    mocaIfClientResetFailNodeMask,
    mocaIfConnectedNodesInfo,
    mocaIfFlowInfo,
    mocaIfLinkDownCount,
    mocaIfLinkUpTime,
    mocaIfLmoNodeID,
    mocaIfMacAddress,
```

```
    mocaIfMaxIngressNodeBw,
    mocaIfMaxEgressNodeBw,
    mocaIfMocaVersion,
    mocaIfName,
    mocaIfNetworkNamePayload,
    mocaIfNetworkState,
    mocaIfNetworkVersion,
    mocaIfNodeID,
    mocaIfNumNodes,
    mocaIfNumPriority,
    mocaIfPduNumber,
    mocaIfPfsPrivacyProvision,
    mocaIfPowerStateCap,
    mocaIfResetCount,
    mocaIfResetReason,
    mocaIfRFChannel,
    mocaIfSoftwareVersion,
    mocaIfStatus,
    mocaIfSupportedBands,
    mocaIfTxGcdRate,
    mocaIfLinkState,
    mocaIfBeaconTxPwr,
    mocaIfBeaconRxPwr,
    mocaIfNcPowerContour,
    mocaIfNcVersion
}
STATUS    current
DESCRIPTION
    "Parameters to monitor the status of the MoCA
    interface."
 ::= { mocaMIBGroups 5 }

mocaIfStatusOptionalGroup OBJECT-GROUP
OBJECTS   {
    mocaIfNcPrivacySupported,
    mocaIfPasswordHash,
    mocaIfPrivacySupported
}
STATUS    current
DESCRIPTION
    "Parameters to monitor the bandwidth of the MoCA
    interface."
 ::= { mocaMIBGroups 6 }

mocaIfStatsGroup OBJECT-GROUP
OBJECTS   {
    mocaIfRxBroadcast,
    mocaIfRxBytes,
    mocaIfRxCorrectedErrors,
    mocaIfRxDrops,
```

```
        mocaIfRxMulticast,
        mocaIfRxPackets,
        mocaIfRxUnicast,
        mocaIfTxBroadcast,
        mocaIfTxBytes,
        mocaIfTxDrops,
        mocaIfTxMulticast,
        mocaIfTxPackets,
        mocaIfTxUnicast
    }
STATUS    current
DESCRIPTION
    "Parameters to monitor the packet statistics and the
    number of flows in the MoCA interface."
 ::= { mocaMIBGroups 7 }

mocaLinkStatsGroup OBJECT-GROUP
OBJECTS   {
    mocaLinkTxDrops,
    mocaLinkTxPackets
}
STATUS    current
DESCRIPTION
    "Parameter to monitor the packet drop statistics per link of
    the MoCA interface."
 ::= { mocaMIBGroups 9 }

mocaNodeGroup OBJECT-GROUP
OBJECTS   {
    mocaNodeAeNumber,
    mocaNodeAggregationSize,
    mocaNodeMacAddress,
    mocaNodeMocaVersion,
    mocaNodePduNumber,
    mocaNodePowerState,
    mocaNodeRxCorrected,
    mocaNodeRxDrops,
    mocaNodeRxPackets,
    mocaNodePDelay
}
STATUS    current
DESCRIPTION
    "Parameters to monitor the status of other nodes
    in the MoCA network."
 ::= { mocaMIBGroups 10 }

mocaP2mpGroup OBJECT-GROUP
OBJECTS   {
    mocaP2mpPhyRates
}

```

```
STATUS    current
DESCRIPTION
    "Parameters to monitor the PHY rate of each link in the
    MoCA network."
 ::= { mocaMIBGroups 12 }

mocaLinkModGroup OBJECT-GROUP
OBJECTS   {
    mocaLinkType,
    mocaScModList,
    mocaNodePower,
    mocaNodeSNR,
    mocaNodePowerReduction
}
STATUS    current
DESCRIPTION
    "Parameters to monitor the modulation of every sub-carrier
    for each link in the MoCA network."
 ::= { mocaMIBGroups 13 }

mocaTrapGroup NOTIFICATION-GROUP
NOTIFICATIONS {
    mocaTrapAbovePhyThreshold,
    mocaTrapBelowPhyThreshold,
    mocaTrapClientResetFail,
    mocaTrapClientResetSuccess,
    mocaTrapConnectedNodesChange,
    mocaTrapNcPrivacySupportedRec,
    mocaTrapPowerStateBcstRec,
    mocaTrapPowerStateM0Nc,
    mocaTrapPowerStateResp,
    mocaTrapPowerStateTrnsReq,
    mocaTrapPowerStateUcstPen,
    mocaTrapPowerStateWupUr,
    mocaTrapLinkFailure
}
STATUS    current
DESCRIPTION
    "Notifications for bandwidth threshold in the MoCA
    network."
 ::= { mocaMIBGroups 14 }

mocaTrapOptionalGroup NOTIFICATION-GROUP
NOTIFICATIONS {
    mocaTrapIfStatusChange,
    mocaTrapLmoStatus
}
STATUS    current
DESCRIPTION
    "Notifications for bandwidth threshold in the MoCA
```

```
network."  
 ::= { mocaMIBGroups 15 }
```

END

## 7 Acknowledgements

This document has been produced by the Multimedia over Coax Alliance (MoCA) Technical Working Group.

## 8 References

- [1] McCloghrie, K., Perkins, D., Schoenwaelder, J., "Conformance Statements for SMIV2", RFC2580, April, 1999. <http://www.rfc-editor.org/rfc/rfc2580.txt> .
- [2] McCloghrie, K., Perkins, D., Schoenwaelder, J., "Structure of Management Information Version 2 (SMIV2)", RFC2579, April, 1999. <http://www.rfc-editor.org/rfc/rfc2579.txt>.
- [3] McCloghrie, K., Perkins, D., Schoenwaelder, J., "Textual Convention for SMIV2", RFC2578, April, 1999. <http://www.rfc-editor.org/rfc/rfc2578.txt>
- [4] "MoCA Enterprise Structure of Management Information", MOCA-SMI-V1.3-20171122, November 22, 2017.
- [5] "MoCA Access MAC/PHY Specification v2.5", MoCA-Access-M/P-SPEC-V2.5-171122, November 2017.
- [6] "MoCA VENDOR ID TABLE v1.1", MoCA-VID-TABLE-V1.1-20180329, March 29, 2018.



## **MoCA Enterprise Structure of Management Information**

**MoCA-SMI-V1.3-20171122**

### **Document Status Sheet**

Document Control Number:	MoCA-SMI-V1.3-20171122
Document Title:	MoCA Enterprise Structure of Management Information
Revision History:	November 22, 2017
Status	Final



## Status of this Memo

This document proposes the MoCA Enterprise Structure of Management Information, and requests discussion and suggestions for improvements. Distribution of this memo is limited to the MoCA membership and further distribution is subject to the MoCA copyright policy.

## Copyright Notice

Copyright © 2008-2017 Multimedia Over Coax Alliance. All Rights Reserved. MoCA is a trademark or registered trademark of the Multimedia Over Coax Alliance in the United States and other countries.

## Abstract

This document specifies the structure of the Simple Network Management Protocol (SNMP) Management Information Base (MIB) sub-tree under the MoCA enterprise number 31621 assigned by IANA [2].

## Table of Contents

1	Introduction .....	2
1.1	Conventions .....	2
2	OID Assignments.....	3
3	MIB Definition .....	3
4	Acknowledgements.....	4
5	References .....	4

## 1 Introduction

Multimedia over Coax Alliance (MoCA) was assigned enterprise number 31621 by IANA [2]. MoCA Alliance owns this enterprise number and may assign any Management Information Base (MIB) Object Identifier (OID) under the MoCA Enterprise OID.

iso.org.dod.internet.private.enterprises.moca (1.3.6.1.4.1.31621)

The assignment of OID under MoCA Enterprise is defined in a MIB definition file.

### 1.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 [1].

## 2 OID Assignments

The following table summarizes the Object Identifier (OID) assigned under the MoCA Enterprise sub-tree. Only two levels of OID are documented here,

- First sub level is the OID under iso.org.dod.internet.private.enterprises.moca (1.3.6.1.4.1.31621)
- Second sub level is the OID under iso.org.dod.internet.private.enterprises.moca.N (1.3.6.1.4.1.31621.N), which N is the first sub level OID.

**Table 1: OID Assignments**

First Sub Level OID	Second Sub Level OID	Object Name	Descriptions
1		mocaMIBs	SNMP MIBs for managed objects in MoCA devices.
1	1	moca11	SNMP MIB for managed objects for MoCA devices that support MoCA 1.1
1	2	moca20	SNMP MIB for managed objects for MoCA devices that support MoCA 2.0.
1	3	moca25	SNMP MIB for managed objects for MoCA devices that support MoCA 2.5
1	4	mocaAccess25	SNMP MIB for managed objects for MoCA devices that support MoCA Access 2.5
2			
3			
4			

## 3 MIB Definition

```
-- *****
-- MoCA-SMI.my: MoCA Enterprise Structure of Management Information
--
-- Copyright (c) 2008 by Multimedia over Coax Alliance
-- All rights reserved.
--
-- *****
--
```

```
MOCA-SMI DEFINITIONS ::= BEGIN
```

```
IMPORTS
    MODULE-IDENTITY,
    OBJECT-IDENTITY,
    enterprises
    FROM SNMPv2-SMI ;
```

Copyright © 2018 by Multimedia over Coax Alliance, (MoCA®).  
 All rights reserved. MoCA®.  
[www.mocalliance.org](http://www.mocalliance.org). [help@mocalliance.org](mailto:help@mocalliance.org)

## MoCA Enterprise Structure of Management Information\_20171122

```
moca MODULE-IDENTITY
  LAST-UPDATED      "200809220000Z" -- September 22, 2008 00:00 AM
  ORGANIZATION      "Multimedia over Coax Alliance (MoCA) Technical
                    Working Group."
  CONTACT-INFO      "www.mocalliance.org

                    Postal:      Multimedia over Coax Alliance
                               2400 Camino Ramon
                               Suite 375
                               San Ramon, CA 94583
                               USA
                    Tel:         +1 925 275-6606
                    Email:       help@mocalliance.org"
  DESCRIPTION        "The Structure of Management Information (SMI) for
                    Multimedia over Coax Alliance (MoCA) enterprise."

  REVISION           "200809220000Z" -- September 22, 2008 00:00 AM
  DESCRIPTION        "Initial version of this MIB module."
  ::= { enterprises 31621 } -- assigned by IANA

mocaMIBs OBJECT-IDENTITY
  STATUS             current
  DESCRIPTION        "mocaMIBs is the root of the MoCA Alliance assigned
                    OID subtree for assignment to MIB modules describing
                    managed objects implemented in MoCA devices that support
                    MoCA MAC/PHY Specification."
  ::= { moca 1 }

END
```

## 4 Acknowledgements

This document has been produced by the Multimedia over Coax Alliance (MoCA®) Technical Work Group.

## 5 References

- [1] Bradner S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997. <<http://www.ietf.org/rfc/rfc2119.txt>>
- [2] IANA, "Private Enterprise Numbers". <<http://www.iana.org/assignments/enterprise-numbers>>.

1  
2  
3  
4  
5  
6  
7  
  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44



**MoCA VENDOR ID TABLE v1.1**

**MoCA-VID-TABLE-V1.1-20180329**

**CONFIDENTIALITY.** This document and each element of this document are the Confidential Information of the Multimedia over Coax Alliance (MoCA) and of the MoCA members that contributed to this document. Both MoCA and/or any such MoCA members may enforce such obligations of confidentiality directly. Your use of this document is subject to your agreement with MoCA, including without limitation the obligations of confidentiality. You may not distribute this document to any person or entity other than as expressly set forth in your Agreement with MoCA. No part of this document may be modified, reproduced, otherwise distributed or displayed, in any form or by any means, in whole or in part, without the prior written permission of MoCA.

Copyright © 2018 Multimedia Over Coax Alliance. All Rights Reserved. MoCA is a trademark or registered trademark of the Multimedia Over Coax Alliance in the United States and other countries.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53

**IMPORTANT NOTICE.** THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". NEITHER MOCA NOR ANY MEMBER OF MOCA MAKES ANY REPRESENTATIONS OR WARRANTIES OF ANY KIND WHATSOEVER WITH RESPECT TO (A) THIS DOCUMENT, (B) ANY PRODUCT THAT IS DEVELOPED OR MANUFACTURED IN ACCORDANCE WITH THE SPECIFICATIONS IN THIS DOCUMENT OR (C) THE INTEROPERABILITY OF ANY SUCH PRODUCT WITH ANY OTHER PRODUCT. MOCA AND MOCA MEMBERS DISCLAIM ALL IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY, NON-INFRINGEMENT AND TITLE. NEITHER MOCA NOR ANY MEMBER OF MOCA MAKES ANY REPRESENTATIONS OR WARRANTIES THAT THE CONTENTS OF THE DOCUMENT ARE COMPLETE, ACCURATE OR SUITABLE FOR ANY PURPOSE OR THAT ANY PRODUCT OR OTHER IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY PATENTS, COPYRIGHTS OR OTHER RIGHTS. IN NO EVENT WILL MOCA OR ANY MOCA MEMBER BE LIABLE FOR ANY LOSSES, INVESTMENTS MADE, LIABILITIES, LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OF DATA, INTERRUPTION OF BUSINESS, OR FOR ANY DIRECT, INDIRECT, SPECIAL OR EXEMPLARY, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONTRACT, TORT, NEGLIGENCE OR OTHER LEGAL THEORY, INCLUDING WITHOUT LIMITATION IN CONNECTION WITH THE USE OF THIS DOCUMENT, THE INFORMATION CONTAINED HEREIN OR ANY PRODUCT OR IMPLEMENTATION, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. USE OF THIS DOCUMENT IS AT YOUR SOLE RISK. From time to time MoCA may issue improvements, enhancements and other changes to the specification described in this document.

1  
2  
3  
4  
5

## Document Status Sheet

<b>Document Control Number:</b>	MoCA-VID-TABLE-V1.1-20180329
<b>Document Title:</b>	MoCA VENDOR ID TABLE v1.1
<b>Revision History:</b>	Issued July 2, 2010 Issued April 5, 2012 Issued February 8, 2013 Issued Mar 29, 2018
<b>Date:</b>	Mar 29, 2018
<b>Status:</b>	<del>Work in Progress</del> <del>Draft</del> Issued
<b>Distribution Restrictions:</b>	MoCA members

6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22

**Key to Document Status Codes:**

- Work in Progress**      An incomplete document, designed to guide discussion and generate feedback, that may include several alternative requirements for consideration.
- Draft**      A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
- Issued**      A stable document, which has undergone rigorous member and vendor review and is suitable for product design and development, cross-vendor interoperability, and for certification testing.

# 1 Introduction & Scope

2  
3 VENDOR\_ID is a 16-bit field present in some MoCA messages to indicate the inclusion of vendor-specific  
4 information. The definition of Vendor ID and how it can be used to send vendor-specific information can be  
5 found in the MoCA specifications. This document captures the most up to date Vendor ID values assigned  
6 by the MoCA Spec Working Group (SWG).  
7  
8 If a MoCA member requires a Vendor ID, the member should send an official request to the SWG reflector  
9 at [moca-kavi-swg@members.mocalliance.org](mailto:moca-kavi-swg@members.mocalliance.org). SWG will process the request and update this document  
10 accordingly.  
11

## 1 2 References

- 2  
3 [1] MoCA-VID-TABLE-V1.0-10172013



### 3 Vendor ID Assignments

The current VENDOR\_ID assignments are shown in Table 3-1. Future VENDOR\_ID values will be assigned from the reserved range.

**Table 3-1: Vendor ID Assignments**

VENDOR_ID Range	Vendor
0x0000	MoCA
0x0001	MaxLinear, Inc.
0x0002 – 0x000F	MoCA
0x0010 – 0x001F	MaxLinear, Inc.
0x0020 – 0x002F	Broadcom Inc.
0x0030 – 0x003F	Not Available [1]
0x0040 – 0x004F	Not Available [1]
0x0050 – 0x005F	ARRIS
0x0060 – 0x006F	Not Available [1]
0x0070 – 0x007F	Intel Corporation
0x0080 – 0x008F	DIRECTV
0x0090	IEEE Std 802.1 AS
0x0091 – 0x009F	Reserved for Future Use
0x00A0 – 0x00AF	ARRIS
0x00B0 – 0x00BF	Complex IQ, Inc.
0x00C0 – 0x00CF	DISH Technologies, LLC
0x00D0 – 0x00DF	InCoax Networks AB
0x00E0 – 0x00EF	Verizon Services Corp.
0x00F0 – 0x00FF	MSTAR Semiconductor
0x0100 – 0x010F	Comcast Corp.
0x0110 – 0x011F	LUSTER, Inc.
0x0120– 0xFFFF	Reserved for Future Use