30.3.1.1.5 aFramesReceivedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>A count of frames that are successfully received (receiveOK). This does not include frames received with frame-too-long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error. This counter is incremented when the ReceiveStatus is reported as receiveOK. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.6 aFrameCheckSequenceErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>A count of frames that are an integral number of octets in length and do not pass the FCS check. This counter is incremented when the ReceiveStatus is reported as frameCheckError. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.7 aAlignmentErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>A count of frames that are not an integral number of octets in length and do not pass the FCS check. This counter is incremented when the ReceiveStatus is reported as alignmentError. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.8 aOctetsTransmittedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 1 230 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>A count of data and padding octets of frames that are successfully transmitted. This counter is incremented when the TransmitStatus is reported as transmitOK. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.9 aFramesWithDeferredXmissions

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 13 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames whose transmission was delayed on its first attempt because the medium was busy. This counter is incremented when the Boolean variable deferred has been asserted by the TransmitLinkMgmt function (4.2.8). Frames involved in any collisions are not counted. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.10 aLateCollisions

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of the times that a collision has been detected later than 512 BT into the transmitted packet. A late collision is counted twice, i.e., both as a collision and as a lateCollision. This counter is incremented when the lateCollisionCount variable is nonzero. The actual update is incremented in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.11 aFramesAbortedDueToXSColls

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 3255 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of the frames that, due to excessive collisions, are not transmitted successfully. This counter is incremented when the value of the attempts variable equals attemptLimit during a transmission. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.12 aFramesLostDueToIntMACXmitError

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that would otherwise be transmitted by the station, but could not be sent due to an internal MAC sublayer transmit error. If this counter is incremented, then none of the other counters in this subclause is incremented. The exact meaning and mechanism for incrementing this counter is implementation dependent.;

30.3.1.1.13 aCarrierSenseErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of times that the carrierSense variable was not asserted or was deasserted during the transmission of a frame without collision (see 7.2.4.6). This counter is incremented when the carrierSenseFailure flag is true at the end of transmission. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.14 aOctetsReceivedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 1 230 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of data and padding octets in frames that are successfully received. This does not include octets in frames received with frame-too-long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error. This counter is incremented when the result of a reception is reported as a receiveOK status. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.15 aFramesLostDueToIntMACRcvError

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that would otherwise be received by the station, but could not be accepted due to an internal MAC sublayer receive error. If this counter is incremented, then none of the other counters in this subclause is incremented. The exact meaning and mechanism for incrementing this counter is implementation dependent.;

30.3.1.1.16 aPromiscuousStatus

ATTRIBUTE

APPROPRIATE SYNTAX:
> BOOLEAN

BEHAVIOUR DEFINED AS:
> A GET operation returns the value "true" for promiscuous mode enabled, and "false" otherwise.
>
> Frames without errors received solely because this attribute has the value "true" are counted as frames received correctly; frames received in this mode that do contain errors update the appropriate error counters.
>
> A SET operation to the value "true" provides a means to cause the LayerMgmtRecognizeAddress function to accept frames regardless of their destination address.
>
> A SET operation to the value "false" causes the MAC sublayer to return to the normal operation of carrying out address recognition procedures for station, broadcast, and multicast group addresses (LayerMgmtRecognizeAddress function).;

30.3.1.1.17 aReadMulticastAddressList

ATTRIBUTE

APPROPRIATE SYNTAX:
> SEQUENCE OF MAC addresses

BEHAVIOUR DEFINED AS:
> The current multicast address list.;

30.3.1.1.18 aMulticastFramesXmittedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that are successfully transmitted, as indicated by the status value transmitOK, to a group destination address other than broadcast. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.19 aBroadcastFramesXmittedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of the frames that were successfully transmitted as indicated by the TransmitStatus transmitOK, to the broadcast address. Frames transmitted to multicast addresses are not broadcast frames and are excluded. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.20 aFramesWithExcessiveDeferral

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 412 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that deferred for an excessive period of time. This counter may only be incremented once per LLC transmission. This counter is incremented when the excessDefer flag is set. The actual update occurs in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.1.21 aMulticastFramesReceivedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that are successfully received and are directed to an active nonbroadcast group address. This does not include frames received with frame-too-long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error. This counter is incremented as indicated by the receiveOK status, and the value in the destinationField. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.22 aBroadcastFramesReceivedOK

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames that are successfully received and are directed to the broadcast group address. This does not include frames received with frame-too-long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error. This counter is incremented as indicated by the receiveOK status, and the value in the destinationField. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.23 aInRangeLengthErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames with a length field value between the minimum unpadded LLC data size and the maximum allowed LLC data size, inclusive, that does not match the number of LLC data octets received. The counter also contains frames with a length field value less than the minimum unpadded LLC data size. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.24 aOutOfRangeLengthField

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames with a length field value greater than the maximum allowed LLC data size. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.25 aFrameTooLongErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 815 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of frames received that exceed the maximum permitted frame size. This counter is incremented when the status of a frame reception is frameTooLong. The actual update occurs in the LayerMgmtReceiveCounters procedure (5.2.4.3).;

30.3.1.1.26 aMACEnableStatus

ATTRIBUTE

APPROPRIATE SYNTAX:
> BOOLEAN

BEHAVIOUR DEFINED AS:
> True if MAC sublayer is enabled and false if disabled. This is accomplished by setting or checking the values of the receiveEnabled and transmitEnabled variables.Setting to true provides a means to cause the MAC sublayer to enter the normal operational state at idle. The PLS is reset by this operation (see 7.2.2.2.1). This is accomplished by setting receiveEnabled and transmitEnabled to true.
> Setting to false causes the MAC sublayer to end all transmit and receive operations, leaving it in a disabled state. This is accomplished by setting receiveEnabled and transmitEnabled to false.;

30.3.1.1.27 aTransmitEnableStatus

ATTRIBUTE

APPROPRIATE SYNTAX:
BOOLEAN

BEHAVIOUR DEFINED AS:
True if transmission is enabled and false otherwise. This is accomplished by setting or checking the value of the transmitEnabled variable.
Setting this to true provides a means to enable MAC sublayer frame transmission (TransmitFrame function). This is accomplished by setting transmitEnabled to true.
Setting this to false will inhibit the transmission of further frames by the MAC sublayer (TransmitFrame function). This is accomplished by setting transmitEnabled to false.;

30.3.1.1.28 aMulticastReceiveStatus

ATTRIBUTE

APPROPRIATE SYNTAX:
BOOLEAN

BEHAVIOUR DEFINED AS:
True if multicast receive is enabled, and false otherwise. Setting this to true provides a means to cause the MAC sublayer to return to the normal operation of multicast frame reception. Setting this to false will inhibit the reception of further multicast frames by the MAC sublayer.;

30.3.1.1.29 aReadWriteMACAddress

ATTRIBUTE

APPROPRIATE SYNTAX:
MACAddress

BEHAVIOUR DEFINED AS:
Read the MAC station address or change the MAC station address to the one supplied (RecognizeAddress function). Note that the supplied station address shall not have the group bit set and shall not be the null address.;

30.3.1.1.30 aCollisionFrames

ATTRIBUTE

APPROPRIATE SYNTAX:
A SEQUENCE of 32 generalized nonresettable counters. Each counter has a maximum increment rate of 13 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
A histogram of collision activity. The indices of this array (1 to attemptLimit – 1) denote the number of collisions experienced in transmitting a frame. Each element of this array contains a counter that denotes the number of frames that have experienced a specific number of collisions. When the TransmitStatus is reported as transmitOK and the value of the attempts variable equals n, then collisionFrames[n–1] counter is incremented. The elements of this array are incremented in the LayerMgmtTransmitCounters procedure (5.2.4.2).;

30.3.1.2 MAC entity actions

30.3.1.2.1 acInitializeMAC

ACTION

APPROPRIATE SYNTAX:
>None required

BEHAVIOUR DEFINED AS:
>This action provides a means to call the Initialize procedure (4.2.7.5). This action also results in the initialization of the PLS.;

30.3.1.2.2 acAddGroupAddress

ACTION

APPROPRIATE SYNTAX:
>MACAddress

BEHAVIOUR DEFINED AS:
>Add the supplied multicast group address to the address recognition filter (RecognizeAddress function).;

30.3.1.2.3 acDeleteGroupAddress

ACTION

APPROPRIATE SYNTAX:
>MACAddress

BEHAVIOUR DEFINED AS:
>Delete the supplied multicast group address from the address recognition filter (RecognizeAddress function).;

30.3.1.2.4 acExecuteSelfTest

ACTION

APPROPRIATE SYNTAX:
>None required

BEHAVIOUR DEFINED AS:
>Execute a self-test and report the results (success or failure). The actual mechanism employed to carry out the self-test is not defined in this standard. If a clause 22 MII is present then this action shall also invoke a data integrity test using MII loopback, returning to normal operation on completion of the test.;

30.3.2 PHY entity managed object class

This subclause formally defines the behaviours for the oPHYEntity managed object class attributes, actions and notifications. Management of that portion of the physical sublayer whose physical containment within the DTE is optional is outside the scope of this clause.

30.3.2.1 PHY entity attributes

30.3.2.1.1 aPHYID

ATTRIBUTE

APPROPRIATE SYNTAX:
    INTEGER

BEHAVIOUR DEFINED AS:
    The value of aPHYID is assigned so as to uniquely identify a PHY, i.e., Physical Layer among the subordinate managed objects of system (systemID and system are defined in ISO/IEC 10165-2: 1992 [SMI], Definition of management information).;

30.3.2.1.2 aPhyType

ATTRIBUTE

APPROPRIATE SYNTAX:
    An ENUMERATED VALUE that has one of the following entries:
    other   Undefined
    unknown   Initializing, true state or type not yet known
    none   MII present and nothing connected
    10 Mb/s   Clause 7 10 Mb/s Manchester
    100BASE-T4   Clause 23 100 Mb/s 8B/6T
    100BASE-X   Clause 24 100 Mb/s 4B/5B

BEHAVIOUR DEFINED AS:
    A read-only value that identifies the PHY type. The enumeration of the type is such that the value matches the clause number of the standard that specifies the particular PHY. The value of this attribute maps to the value of aMAUType. The enumeration "none" can only occur in a standard implementation an MII exists and there is nothing connected. However, the attribute aMIIDetect should be used to determine whether an MII exists or not.;

30.3.2.1.3 aPhyTypeList

ATTRIBUTE

APPROPRIATE SYNTAX:
    A SEQUENCE that meets the requirements of the description below:
    other   Undefined
    unknown   Initializing, true state or type not yet known
    none   MII present and nothing connected
    10 Mb/s   Clause 7 10 Mb/s Manchester
    100BASE-T4   Clause 23 100 Mb/s 8B/6T
    100BASE-X   Clause 24 100 Mb/s 4B/5B

BEHAVIOUR DEFINED AS:
    A read-only list of the possible types that the PHY could be, identifying the ability of the PHY. If clause 28, Auto-Negotiation, is present, then this attribute will map to the local technology ability.;

30.3.2.1.4 aSQETestErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
    Generalized nonresettable counter. This counter has a maximum increment rate of 16 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> A count of times that the SQE_TEST_ERROR was received. The SQE_TEST_ERROR is set in accordance with the rules for verification of the SQE detection mechanism in the PLS Carrier Sense function (see 7.2.4.6). The SQE test function is not a part of 100 Mb/s PHY operation, and so SQETestErrors will not occur in 100 Mb/s PHYs.;

30.3.2.1.5 aSymbolErrorDuringCarrier

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 160 000 counts per second for 100 Mb/s implementations

BEHAVIOUR DEFINED AS:
> A count of the number of times when valid carrier was present and there was at least one occurrence of an invalid data symbol. This can increment only once per valid carrier event. If a collision is present this attribute will not increment.;

30.3.2.1.6 aMIIDetect

ATTRIBUTE

APPROPRIATE SYNTAX:
> An ENUMERATED VALUE that has one of the following entries:
> unknown
> present, nothing connected
> present, connected
> absent

BEHAVIOUR DEFINED AS:
> An attribute of the PhyEntity managed object class indicating whether an MII connector is physically present, and if so whether it is detectably connected as specified in 22.2.2.12.;

30.3.2.1.7 aPhyAdminState

ATTRIBUTE

APPROPRIATE SYNTAX:
> An ENUMERATED VALUE that has the following entries:
> disabled
> enabled

BEHAVIOUR DEFINED AS:
> A disabled PHY neither transmits nor receives. The PHY shall be explicitly enabled to restore operation. The acPhyAdminControl action provides this ability. The port enable/disable function as reported by this attribute is preserved across DTE reset including loss of power. Only one PHY per MAC can be enabled at any one time. Setting a PHY to the enabled state using the action acPhyAdminControl will result in all other instances of PHY (indicated by PhyID) instantiated within the same MAC to be disabled. If a clause 22 MII is present then setting this attribute to "disable" will result in electrical isolation as defined in 22.2.4.1.6, Isolate; and setting this attribute to "enabled" will result in normal operation as defined in 22.2.4.1.5, Power down; and 22.2.4.1.6, Isolate.;

30.3.2.2 PHY entity actions

30.3.2.2.1 acPhyAdminControl

ACTION

APPROPRIATE SYNTAX:
      Same as aPortAdminState

BEHAVIOUR DEFINED AS:
      This action provides a means to alter aPhyAdminState. Setting a PHY to the enabled state will
      result in all other instances of PHY being disabled.;

30.4 Layer management for 10 Mb/s and 100 Mb/s baseband repeaters

30.4.1 Repeater managed object class

This subclause formally defines the behaviours for the oRepeater managed object class, attributes, actions, and notifications.

30.4.1.1 Repeater attributes

30.4.1.1.1 aRepeaterID

ATTRIBUTE

APPROPRIATE SYNTAX:
      INTEGER

BEHAVIOUR DEFINED AS:
      The value of aRepeaterID is assigned so as to uniquely identify a repeater among the subordinate
      managed objects of system (systemID and system are defined in ISO/IEC 10165-2: 1992 [SMI],
      Definition of management information).;

30.4.1.1.2 aRepeaterType

ATTRIBUTE

APPROPRIATE SYNTAX:
      An INTEGER that meets the requirements of the description below:

9	10 Mb/s Baseband
271	100 Mb/s Baseband, Class I
272	100 Mb/s Baseband, Class II
other	See 20.2.2.3
unknown	Initializing, true state or type not yet known

BEHAVIOUR DEFINED AS:
      Returns a value that identifies the CSMA/CD repeater type. The enumeration of the type is such
      that the value matches the clause number of the standard that specifies the particular repeater, with
      further numerical identification for the repeater classes within the same clause.;

30.4.1.1.3 aRepeaterGroupCapacity

ATTRIBUTE

APPROPRIATE SYNTAX:
        INTEGER

BEHAVIOUR DEFINED AS:
        The aRepeaterGroupCapacity is the number of groups that can be contained within the repeater.
        Within each managed repeater, the groups are uniquely numbered in the range from 1 to
        aRepeaterGroupCapacity.

        Some groups may not be present in a given repeater instance, in which case the actual number of
        groups present is less than aRepeaterGroupCapacity. The number of groups present is never
        greater than aRepeaterGroupCapacity.;

30.4.1.1.4 aGroupMap

ATTRIBUTE

APPROPRIATE SYNTAX:
        BITSTRING

BEHAVIOUR DEFINED AS:
        A string of bits which reflects the current configuration of units that are viewed by group managed
        objects. The length of the bitstring is "aRepeaterGroupCapacity" bits. The first bit relates to group
        1. A "1" in the bitstring indicates presence of the group, "0" represents absence of the group.;

30.4.1.1.5 aRepeaterHealthState

ATTRIBUTE

APPROPRIATE SYNTAX:
        An ENUMERATED VALUE LIST that has the following entries:
        other   undefined or unknown
        ok   no known failures
        repeaterFailure known to have a repeater related failure
        groupFailure   known to have a group related failure
        portFailure   known to have a port related failure
        generalFailure has a failure condition, unspecified type

BEHAVIOUR DEFINED AS:
        The aRepeaterHealthState attribute indicates the operational state of the repeater. The
        aRepeaterHealthData and aRepeaterHealthText attributes may be consulted for more specific
        information about the state of the repeater's health. In case of multiple kinds of failures (e.g.,
        repeater failure and port failure), the value of this attribute shall reflect the highest priority in the
        following order:
        repeater failure
        group failure
        port failure
        general failure;

30.4.1.1.6 aRepeaterHealthText

ATTRIBUTE

APPROPRIATE SYNTAX:
A PrintableString, 255 characters max

BEHAVIOUR DEFINED AS:
The aRepeaterHealthText attribute is a text string that provides information relevant to the operational state of the repeater. Repeater vendors may use this mechanism to provide detailed failure information or instructions for problem resolution.

The contents are vendor specific.;

30.4.1.1.7 aRepeaterHealthData

ATTRIBUTE

APPROPRIATE SYNTAX:
OCTET STRING, 0–255

BEHAVIOUR DEFINED AS:
The aRepeaterHealthData attribute is a block of data octets that provides information relevant to the operational state of the repeater. The encoding of this data block is vendor dependent. Repeater vendors may use this mechanism to provide detailed failure information or instructions for problem resolution.;

30.4.1.1.8 aTransmitCollisions

ATTRIBUTE

APPROPRIATE SYNTAX:
Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
For a clause 9 repeater, the counter increments every time the repeater state diagram enters the TRANSMIT COLLISION state from any state other than ONE PORT LEFT (figure 9-2). For a clause 27 repeater, the counter increments every time the Repeater Core state diagram enters the JAM state as a result of Activity(ALL) > 1 (figure 27-2).;

30.4.1.2 Repeater actions

30.4.1.2.1 acResetRepeater

ACTION

APPROPRIATE SYNTAX:
None required

BEHAVIOUR DEFINED AS:
This causes a transition to the START state of figure 9-2 for a clause 9 repeater, or to the START state of figure 27-2 for a clause 27 repeater. The repeater performs a disruptive self-test that has the following characteristics:
1. The components are not specified
2. The test resets the repeater but without affecting management information about the repeater
3. The test does not inject packets onto any segment
4. Packets received during the test may or may not be transferred
5. The test does not interfere with management functions
This causes a nRepeaterReset notification to be sent.;

30.4.1.2.2 acExecuteNonDisruptiveSelfTest

ACTION

APPROPRIATE SYNTAX:
 None required

BEHAVIOUR DEFINED AS:
 The repeater performs a vendor-specific, non-disruptive self-test that has the following
 characteristics:
 1. The components are not specified
 2. The test does not change the state of the repeater or management information about the repeater
 3. The test does not inject packets onto any segment
 4. The test does not prevent the transfer of any packets
 5. Completion of the test causes a nRepeaterHealth to be sent.;

30.4.1.3 Repeater notifications

30.4.1.3.1 nRepeaterHealth

NOTIFICATION

APPROPRIATE SYNTAX:
 A SEQUENCE of three data types. The first is mandatory, the following two are optional. The first
 is the value of the attribute aRepeaterHealthState. The second is the value of the attribute
 aRepeaterHealthText. The third is the value of the attribute aRepeaterHealthData

BEHAVIOUR DEFINED AS:
 This notification conveys information related to the operational state of the repeater. See the
 aRepeaterHealthState, aRepeaterHealthText, and aRepeaterHealthData attributes for descriptions
 of the information that is sent.

 The nRepeaterHealth notification is sent only when the health state of the repeater changes. The
 nRepeaterHealth notification shall contain repeaterHealthState. repeaterHealthData and
 repeaterHealthText may or may not be included. The nRepeaterHealth notification is not sent as a
 result of powering up a repeater.;

30.4.1.3.2 nRepeaterReset

NOTIFICATION

APPROPRIATE SYNTAX:
 A SEQUENCE of three data types. The first is mandatory, the following two are optional. The first
 is the value of the attribute aRepeaterHealthState. The second is the value of the attribute
 aRepeaterHealthText. The third is the value of the attribute aRepeaterHealthData

BEHAVIOUR DEFINED AS:
 This notification conveys information related to the operational state of the repeater. The
 nRepeaterReset notification is sent when the repeater is reset as the result of a power-on condition
 or upon completion of the acResetRepeater action. The nRepeaterReset notification shall contain
 repeaterHealthState. repeaterHealthData and repeaterHealthText may or may not be included.;

30.4.1.3.3 nGroupMapChange

NOTIFICATION

APPROPRIATE SYNTAX:
 BITSTRING

BEHAVIOUR DEFINED AS:
> This notification is sent when a change occurs in the group structure of a repeater. This occurs only when a group is logically removed from or added to a repeater. The nGroupMapChange notification is not sent when powering up a repeater. The value of the notification is the updated value of the aGroupMap attribute.;

30.4.2 Group managed object class

This subclause formally defines the behaviours for the oGroup managed object class, attributes, actions, and notifications.

30.4.2.1 Group attributes

30.4.2.1.1 aGroupID

ATTRIBUTE

APPROPRIATE SYNTAX:
> INTEGER

BEHAVIOUR DEFINED AS:
> A value unique within the repeater. The value of aGroupID is assigned so as to uniquely identify a group among the subordinate managed objects of the containing object (oRepeater). This value is never greater than aRepeaterGroupCapacity.;

30.4.2.1.2 aGroupPortCapacity

ATTRIBUTE

APPROPRIATE SYNTAX:
> INTEGER

BEHAVIOUR DEFINED AS:
> The aGroupPortCapacity is the number of ports contained within the group. Valid range is 1–1024. Within each group, the ports are uniquely numbered in the range from 1 to aGroupPortCapacity. Some ports may not be present in a given group instance, in which case the actual number of ports present is less than aGroupPortCapacity. The number of ports present is never greater than aGroupPortCapacity.;

30.4.2.1.3 aPortMap

ATTRIBUTE

APPROPRIATE SYNTAX:
> BitString

BEHAVIOUR DEFINED AS:
> A string of bits that reflects the current configuration of port managed objects within this group. The length of the bitstring is "aGroupPortCapacity" bits. The first bit relates to group 1. A "1" in the bitstring indicates presence of the port, "0" represents absence of the port.;

30.4.2.2 Group notifications

30.4.2.2.1 nPortMapChange

NOTIFICATION

APPROPRIATE SYNTAX:
> BitString

BEHAVIOUR DEFINED AS:
> This notification is sent when a change occurs in the port structure of a group. This occurs only when a port is logically removed from or added to a group. The nPortMapChange notification is not sent when powering up a repeater. The value of the notification is the updated value of the aPortMap attribute.;

30.4.3 Repeater port managed object class

This subclause formally defines the behaviours for the oRepeaterPort managed object class, attributes, actions, and notifications.

30.4.3.1 Port attributes

30.4.3.1.1 aPortID

ATTRIBUTE

APPROPRIATE SYNTAX:
> INTEGER

BEHAVIOUR DEFINED AS:
> A value unique in the group. It is assumed that ports are partitioned into groups that also have IDs. The value of aPortID is assigned so as to uniquely identify a repeater port among the subordinate managed objects of the containing object (oGroup). This value can never be greater than aGroupPortCapacity.;

30.4.3.1.2 aPortAdminState

ATTRIBUTE

APPROPRIATE SYNTAX:
> An ENUMERATED VALUE LIST that has the following entries:
> disabled
> enabled

BEHAVIOUR DEFINED AS:
> A disabled port neither transmits nor receives. The port shall be explicitly enabled to restore operation. The acPortAdminControl action provides this ability. The port enable/disable function as reported by this attribute is preserved across repeater reset including loss of power. aPortAdminState takes precedence over auto-partition and functionally operates between the auto-partition mechanism and the AUI/PMA. Auto-partition is reinitialized whenever acPortAdminControl is enabled.;

30.4.3.1.3 aAutoPartitionState

ATTRIBUTE

APPROPRIATE SYNTAX:
> An ENUMERATED VALUE LIST that has the following entries:
> autoPartitioned
> notAutoPartitioned

BEHAVIOUR DEFINED AS:
> The aAutoPartitionState flag indicates whether the port is currently partitioned by the repeater's auto-partition protection. The conditions that cause port partitioning are specified in partition state diagram in clauses 9 and 27. They are not differentiated here. A clause 27 repeater port partitions on entry to the PARTITION WAIT state of the partition state diagram (figure 27-8).;

30.4.3.1.4 aReadableFrames

ATTRIBUTE

APPROPRIATE SYNTAX:
>   Generalized nonresettable counter. This counter has a maximum increment rate of 15 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>   A representation of the total frames of valid frame length. Increment counter by one for each frame whose OctetCount is greater than or equal to minFrameSize and less than or equal to maxFrameSize (see 4.4.2.1) and for which the FCSError and CollisionEvent signals are not asserted.

>   NOTE—This statistic provides one of the parameters necessary for obtaining the packet error rate.;

30.4.3.1.5 aReadableOctets

ATTRIBUTE

APPROPRIATE SYNTAX:
>   Generalized nonresettable counter. This counter has a maximum increment rate of 1 240 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>   Increment counter by OctetCount for each frame which has been determined to be a readable frame.

>   NOTE—This statistic provides an indicator of the total data transferred.;

30.4.3.1.6 aFrameCheckSequenceErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
>   Generalized nonresettable counter. This counter has a maximum increment rate of 15 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>   Increment counter by one for each frame with the FCSError signal asserted and the FramingError and CollisionEvent signals deasserted and whose OctetCount is greater than or equal to minFrameSize and less than or equal to maxFrameSize (see 4.4.2.1).;

30.4.3.1.7 aAlignmentErrors

ATTRIBUTE

APPROPRIATE SYNTAX:
>   Generalized nonresettable counter. This counter has a maximum increment rate of 15 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>   Increment counter by one for each frame with the FCSError and FramingError signals asserted and CollisionEvent signal deasserted and whose OctetCount is greater than or equal to minFrameSize and less than or equal to maxFrameSize (see 4.4.2.1). If aAlignmentErrors is incremented then the aFrameCheckSequenceErrors attribute shall not be incremented for the same frame.;

30.4.3.1.8 aFramesTooLong

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 815 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> Increment counter by one for each frame whose OctetCount is greater than maxFrameSize (see 4.4.2.1). If aFrameTooLong is counted then neither the aAlignmentErrors nor the aFrameCheckSequenceErrors attribute shall be incremented for the frame.;

30.4.3.1.9 aShortEvents

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> Increment counter by one for each CarrierEvent with ActivityDuration less than ShortEventMaxTime. In the 10 Mb/s case ShortEventMaxTime is greater than 74 BT and less than 82 BT. ShortEventMaxTime has tolerances included to provide for circuit losses between a conformance test point at the AUI and the measurement point within the state diagram. In the 100 Mb/s case ShortEventMaxTime is 84 bits (21 nibbles).
>
> NOTES
> 1—shortEvents may indicate externally generated noise hits that will cause the repeater to transmit Runts to its other ports, or propagate a collision (which may be late) back to the transmitting DTE and damaged frames to the rest of the network.
>
> 2—Implementors may wish to consider selecting the ShortEventMaxTime towards the lower end of the allowed tolerance range to accommodate bit losses suffered through physical channel devices not budgeted for within this standard.
>
> 3—Note also that the significance of this attribute is different in 10 and 100 Mb/s collision domains. Clause 9 repeaters perform fragment extension of short events that would be counted as runts on the interconnect ports of other repeaters. Clause 27 repeaters do not perform fragment extension.;

30.4.3.1.10 aRunts

ATTRIBUTE

APPROPRIATE SYNTAX:
> Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
> Increment counter by one for each CarrierEvent that meets one of the following two conditions. Only one test need be made. a) The ActivityDuration is greater than ShortEventMaxTime and less than ValidPacketMinTime and the CollisionEvent signal is de-asserted (10 Mb/s operation) or the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8) has not been entered (100 Mb/s operation). b) The OctetCount is less than 64, the ActivityDuration is greater than ShortEventMaxTime, and the CollisionEvent signal is de-asserted (10 Mb/s operation), or the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8) has not been entered (100 Mb/s operation). ValidPacketMinTime is greater than or equal to 552 BT and less than 565 BT. At 10 Mb/s an event whose length is greater than 74 BT but less than 82 BT shall increment either the aShortEvents attribute or the aRunts attribute, but not both. A CarrierEvent greater than or equal to 552 BT but less than 565 BT may or may not be counted as a runt. ValidPacketMinTime has tolerances included to provide for circuit losses between a conformance test point at the AUI and the measurement point within the state diagram.
>
> NOTE—Runts usually indicate collision fragments, a normal network event.
> In certain situations associated with large diameter networks a percentage of runts may exceed ValidPacket-MinTime.;

30.4.3.1.11 aCollisions

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>For a clause 9 repeater the counter increments for any CarrierEvent signal on any port in which the CollisionEvent signal on this port is asserted. For a clause 27 repeater port the counter increments on entering the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8).;

30.4.3.1.12 aLateEvents

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 75 000 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>For a clause 9 repeater port this counter increments for each CarrierEvent in which the CollIn(X) variable transitions to the value SQE (see 9.6.6.2) while the ActivityDuration is greater than the LateEventThreshold. For a clause 27 repeater port this counter increments on entering the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8) while the ActivityDuration is greater than the LateEventThreshold. Such a CarrierEvent is counted twice, as both a aCollision and as a aLateEvent.
>The LateEventThreshold is greater than 480 BT and less than 565 BT. LateEventThreshold has tolerances included to permit an implementation to build a single threshold to serve as both the LateEventThreshold and ValidPacketMinTime threshold.;

30.4.3.1.13 aVeryLongEvents

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter. This counter has a maximum increment rate of 250 counts per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
>For a clause 9 repeater port this counter increments for each CarrierEvent whose ActivityDuration is greater than the MAU Jabber Lockup Protection timer TW3 (see 9.6.1, 9.6.5). For a clause 27 repeater port this counter increments on entry to the RX JABBER state of the receive timer state diagram (figure 27-7). Other counters may be incremented as appropriate.;

30.4.3.1.14 aDataRateMismatches

ATTRIBUTE

APPROPRIATE SYNTAX:
>Generalized nonresettable counter

BEHAVIOUR DEFINED AS:
Increment counter by one for each frame received by this port that meets all of the conditions required by only one of the following two measurement methods: Measurement method A: 1) The CollisionEvent signal is not asserted (10 Mb/s operation) or the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8) has not been entered (100 Mb/s operation). 2) The ActivityDuration is greater than ValidPacketMinTime. 3) The frequency (data rate) is detectably mismatched from the local transmit frequency. Measurement method B: 1) The CollisionEvent signal is not asserted (10 Mb/s operation) or the COLLISION COUNT INCREMENT state of the partition state diagram (figure 27-8) has not been entered (100 Mb/s operation). 2) The OctetCount is greater than 63. 3) The frequency (data rate) is detectably mismatched from the local transmit frequency. The exact degree of mismatch is vendor specific and is to be defined by the vendor for conformance testing.
When this event occurs, other counters whose increment conditions were satisfied may or may not also be incremented, at the implementor's discretion.

NOTE—Whether or not the repeater was able to maintain data integrity is beyond the scope of this standard.;

30.4.3.1.15 aAutoPartitions

ATTRIBUTE

APPROPRIATE SYNTAX:
Generalized nonresettable counter

BEHAVIOUR DEFINED AS:
Increment counter by one for each time that the repeater has automatically partitioned this port. The conditions that cause a clause 9 repeater port to partition are specified in the partition state diagram in clause 9. They are not differentiated here. A clause 27 repeater port partitions on entry to the PARTITION WAIT state of the partition state diagram (figure 27-8).;

30.4.3.1.16 aIsolates

ATTRIBUTE

APPROPRIATE SYNTAX:
Generalized nonresettable counter. This counter has a maximum increment rate of 400 counts per second at 100 Mb/s

BEHAVIOUR DEFINED AS:
Increment counter by one each time that the repeater port automatically isolates as a consequence of false carrier events. The conditions that cause a port to automatically isolate are as defined by the transition from the FALSE CARRIER state to the LINK UNSTABLE state of the carrier integrity state diagram (figure 27-9).

NOTE—Isolates do not affect the value of aPortAdminState.;

30.4.3.1.17 aSymbolErrorDuringPacket

ATTRIBUTE

APPROPRIATE SYNTAX:
Generalized nonresettable counter. This counter has a maximum increment rate of 160 000 counts per second for 100 Mb/s implementations

BEHAVIOUR DEFINED AS:
A count of the number of times when valid length packet was received at the port and there was at least one occurrence of an invalid data symbol. This can increment only once per valid carrier event. A collision presence at any port of the repeater containing port N, will not cause this attribute to increment.;

30.4.3.1.18 aLastSourceAddress

ATTRIBUTE

APPROPRIATE SYNTAX:
    MACAddress

BEHAVIOUR DEFINED AS:
    The Source Address of the last readable Frame received by this port.;

30.4.3.1.19 aSourceAddressChanges

ATTRIBUTE

APPROPRIATE SYNTAX:
    Generalized nonresettable counter. This counter has a maximum increment rate of 15 000 counts
    per second at 10 Mb/s

BEHAVIOUR DEFINED AS:
    Increment counter by one each time that the aLastSourceAddress attribute has changed.

    NOTE—This may indicate whether a link is connected to a single DTE or another multi-user segment.;

30.4.3.2 Port actions

30.4.3.2.1 acPortAdminControl

ACTION

APPROPRIATE SYNTAX:
    Same as aPortAdminState

BEHAVIOUR DEFINED AS:
    This action provides a means to alter aPortAdminState and exert a BEGIN on the Partitioning state
    diagram (figure 9-6) or the Partition state diagram (figure 27-8) upon taking the value "enabled".;

30.5 Layer management for 10 Mb/s and 100 Mb/s MAUs

30.5.1 MAU managed object class

This subclause formally defines the behaviours for the oMAU managed object class, attributes, actions, and
notifications.

30.5.1.1 MAU attributes

30.5.1.1.1 aMAUID

ATTRIBUTE

APPROPRIATE SYNTAX:
    INTEGER

BEHAVIOUR DEFINED AS:
    The value of aMAUID is assigned so as to uniquely identify a MAU among the subordinate
    managed objects of the containing object.;

30.5.1.1.2 aMAUType

ATTRIBUTE

APPROPRIATE SYNTAX:
> A GET-SET ENUMERATION that meets the requirements of the description below:

global	Reserved for future use
other	See 20.2.2.3
unknown	Initializing, true state or type not yet known
AUI	no internal MAU, view from AUI
10BASE5	Thick coax MAU as specified in clause 8
FOIRL	FOIRL MAU as specified in 9.9
10BASE2	Thin coax MAU as specified in clause 10
10BROAD36	Broadband DTE MAU as specified in clause 11
10BASE-T	UTP MAU as specified in clause 14
10BASE-FP	Passive fiber MAU as specified in clause 16
10BASE-FB	Synchronous fiber MAU as specified in clause 17
10BASE-FL	Asynchronous fiber MAU as specified in clause 18
100BASE-T4	Four-pair Category 3 UTP as specified in clause 23
100BASE-TX	Two-pair Category 5 UTP as specified in clause 25
100BASE-FX	X fiber over PMD as specified in clause 26
802.9a	Integrated services MAU as specified in IEEE Std 802.9 ISLAN-16T

BEHAVIOUR DEFINED AS:
> Returns a value that identifies the 10 Mb/s or 100 Mb/s internal MAU type. The enumeration of the type is such that the value matches the clause number of the standard that specifies the particular MAU. If an AUI is to be identified to access an external MAU, then type "AUI" is returned. A SET operation to one of the possible enumerations indicated by aMAUTypeList will force the MAU into the new operating mode. If a clause 22 MII is present, then this will map to the mode force bits specified in 22.2.4.1. If clause 28, Auto-Negotiation, is operational, then this will change the advertised ability to the single enumeration specified in the SET operation, and cause an immediate link renegotiation. A change in MAU type will also be reflected in oPHYType.;

30.5.1.1.3 aMAUTypeList

ATTRIBUTE

APPROPRIATE SYNTAX:
> A SEQUENCE of ENUMERATIONS that match the syntax of aMAUType

BEHAVIOUR DEFINED AS:
> A GET attribute that returns the possible types that the MAU could be, identifying the ability of the MAU. If clause 28 Auto-Negotiation is present, then this attribute will map to the local technology ability. This attribute maps to aPHYTypeList.;

30.5.1.1.4 aMediaAvailable

ATTRIBUTE

APPROPRIATE SYNTAX:
> An ENUMERATED value list that has the following entries:

other	undefined
unknown	initializing, true state not yet known
available	link or light normal, loopback normal
not available	link loss or low light, no loopback
remote fault	remote fault with no detail
invalid signal	invalid signal, applies only to 10BASE-FB
remote jabber	remote fault, reason known to be jabber
remote link loss	remote fault, reason known to be far-end link loss
remote test	remote fault, reason known to be test

BEHAVIOUR DEFINED AS:

    If the MAU is a link or fiber type (FOIRL, 10BASE-T, 10BASE-F), then this is equivalent to the link test fail state/low light function. For an AUI , 10BASE2, 10BASE5, or 10BROAD36 MAU, this indicates whether or not loopback is detected on the DI circuit. The value of this attribute persists between packets for MAU types AUI, 10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP.

    At power-up or following a reset, the value of this attribute will be "unknown" for AUI, 10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP MAUs. For these MAUs loopback will be tested on each transmission during which no collision is detected. If DI is receiving *input* when DO returns to IDL after a transmission and there has been no collision during the transmission, then loopback will be detected. The value of this attribute will only change during noncollided transmissions for AUI, 10BASE2, 10BASE5, 10BROAD36, and 10BASE-FP MAUs.

    For 100BASE-T4, 100BASE-TX, and 100BASE-FX the enumerations match the states within the respective link integrity state diagrams, figure 23-12 and 24-15. Any MAU that implements management of clause 28 Auto-Negotiation will map remote fault indication to MediaAvailable remote fault.

    The enumeration "remote fault" applies to 10BASE-FB, 100BASE-X, far-end fault indication and non-specified remote faults from a system running clause 28 Auto-Negotiation. The enumerations "remote jabber," "remote link loss," or "remote test" should be used instead of "remote fault" where the reason for remote fault is identified in the remote signaling protocol.

    Where a clause 22 MII is present, a logic one in the remote fault bit (22.2.4.2.9) maps to the enumeration "remote fault," a logic zero in the link status bit (22.2.4.2.11) maps to the enumeration "not available." The enumeration "not available" takes precedence over "remote fault.";

30.5.1.1.5 aLoseMediaCounter

ATTRIBUTE

APPROPRIATE SYNTAX:

    Generalized nonresettable counter. This counter has a maximum increment rate of 10 counts per second

BEHAVIOUR DEFINED AS:

    Counts the number of times that the MAU leaves MediaAvailState "available." Mandatory for MAU type "AUI," optional for all others.;

30.5.1.1.6 aJabber

ATTRIBUTE

APPROPRIATE SYNTAX:

    A SEQUENCE of two indications. The first, JabberFlag, consists of an ENUMERATED value list that has the following entries:

other	undefined
unknown	initializing, true state not yet known
normal	state is true or normal
fault	state is false, fault, or abnormal

    The second, jabberCounter, is a generalized nonresettable counter. This counter has a maximum increment rate of 40 counts per second

BEHAVIOUR DEFINED AS:

    If the MAU is in the JABBER state, the jabberFlag portion of the attribute is set to the "fault" value. The jabberCounter portion of the attribute is incremented each time the flag is set to the "fault" value. This attribute returns the value "other" for type AUI. Note that this counter will not increment for a 100 Mb/s PHY, as there is no defined JABBER state.;

30.5.1.1.7 aMAUAdminState

ATTRIBUTE

APPROPRIATE SYNTAX:
An ENUMERATED value list that has the following entries:
other   undefined
unknown   initializing, true state not yet known
operational   powered and connected
standby   inactive but on
shutdown   similar to power down

BEHAVIOUR DEFINED AS:
A MAU in management state "standby" forces DI and CI to idle and the media transmitter to idle or fault, if supported. The management state "standby" only applies to link type MAUs. The state of MediaAvailable is unaffected. A MAU or AUI in the management state "shutdown" assumes the same condition on DI, CI and the media transmitter as if it were powered down or not connected. For an AUI, this management state will remove power from the AUI. The MAU may return the value "undefined" for Jabber and MediaAvailable attributes when it is in this management state. A MAU in the management state "operational" is fully functional, and operates and passes signals to its attached DTE or repeater port in accordance with its specification.;

30.5.1.1.8 aBbMAUXmitRcvSplitType

ATTRIBUTE

APPROPRIATE SYNTAX:
An ENUMERATED value list that has the following entries:
other   undefined
single   single-cable system
dual   dual-cable system, offset normally zero

BEHAVIOUR DEFINED AS:
Returns a value that indicates the type of frequency multiplexing/cabling system used to separate the transmit and receive paths for the 10BROAD36 MAU. All other types return "undefined.";

30.5.1.1.9 aBroadbandFrequencies

ATTRIBUTE

APPROPRIATE SYNTAX:
A SEQUENCE of two instances of the type INTEGER.

The first INTEGER represents the Transmitter Carrier Frequency. The value of its INTEGER represents the frequency of the carrier divided by 250 kHz.

The second INTEGER represents the Translation Offset Frequency. The value of its INTEGER represents the frequency of the offset divided by 250 kHz

BEHAVIOUR DEFINED AS:
Returns a value that indicates the transmit carrier frequency and translation offset frequency in MHz/4 for the 10BROAD36 MAU. This allows the frequencies to be defined to a resolution of 250 kHz.;

30.5.1.1.10 aFalseCarriers

ATTRIBUTE

APPROPRIATE SYNTAX:
Generalized nonresettable counter. This counter has a maximum increment rate of 160 000 counts per second under maximum network load, and 10 counts per second under zero network load, for 100 Mb/s implementations

BEHAVIOUR DEFINED AS:
>A count of the number of false carrier events during IDLE in 100BASE-X links. This counter does not increment at the symbol rate. It can increment after a valid carrier completion at a maximum rate of once per 100 ms until the next carrier event.;

30.5.1.2 MAU actions

30.5.1.2.1 acResetMAU

ACTION

APPROPRIATE SYNTAX:
>None required

BEHAVIOUR DEFINED AS:
>Resets the MAU in the same manner as would a power-off, power-on cycle of at least 0.5 s duration. During the 0.5 s DO, DI, and CI should be idle.;

30.5.1.2.2 acMAUAdminControl

ACTION

APPROPRIATE SYNTAX:
>The same as used for aMAUAdminState

BEHAVIOUR DEFINED AS:
>Executing an acMAUAdminControl action causes the MAU to assume the aMAUAdminState attribute value of one of the defined valid management states for control input. The valid inputs are "standby," "operational," and "shutdown" state (see the behaviour definition bMAUAdminState for the description of each of these states) except that a "standby" action to a mixing type MAU or an AUI will cause the MAU to enter the "shutdown" management state.;

30.5.1.3 MAU notifications

30.5.1.3.1 nJabber

NOTIFICATION

APPROPRIATE SYNTAX:
>The same as used for aJabber

BEHAVIOUR DEFINED AS:
>The notification is sent whenever a managed MAU enters the JABBER state.;

30.6 Management for link Auto-Negotiation

30.6.1 Auto-Negotiation managed object class

This subclause formally defines the behaviours for the oAuto-Negotiation managed object class, attributes, actions, and notifications.

30.6.1.1 Auto-Negotiation attributes

30.6.1.1.1 aAutoNegID

ATTRIBUTE

APPROPRIATE SYNTAX:
 INTEGER

BEHAVIOUR DEFINED AS:
 The value of aAutoNegID is assigned so as to uniquely identify an Auto-Negotiation managed
 object among the subordinate managed objects of the containing object.;

30.6.1.1.2 aAutoNegAdminState

ATTRIBUTE

APPROPRIATE SYNTAX:
 An ENUMERATED VALUE that has one of the following entries:
 enabled
 disabled

BEHAVIOUR DEFINED AS:
 An interface which has Auto-Negotiation signaling ability will be enabled to do so when this
 attribute is in the enabled state. If disabled then the interface will act as it would if it had no Auto-
 Negotiation signaling. Under these conditions it will immediately be forced to the states indicated
 by a write to the attribute aMAUType.;

30.6.1.1.3 aAutoNegRemoteSignaling

ATTRIBUTE

APPROPRIATE SYNTAX:
 An ENUMERATED VALUE that has one of the following entries:
 detected
 notdetected

BEHAVIOUR DEFINED AS:
 The value indicates whether the remote end of the link is operating Auto-Negotiation signaling or
 not. It shall take the value detected if, during the previous link negotiation, FLP Bursts were
 received from the remote end.;

30.6.1.1.4 aAutoNegAutoConfig

ATTRIBUTE

APPROPRIATE SYNTAX:
 An ENUMERATED VALUE that has one of the following entries:
 other
 configuring
 complete
 disabled
 parallel detect fail

BEHAVIOUR DEFINED AS:
 Indicates whether Auto-Negotiation signaling is in progress or has completed. The enumeration
 "parallel detect fail" maps to a failure in parallel detection as defined in 28.2.3.1.;

30.6.1.1.5 aAutoNegLocalTechnologyAbility

ATTRIBUTE

APPROPRIATE SYNTAX:
>A SEQUENCE that meets the requirements of the description below:
>| | |
>|---|---|
>| global | Reserved for future use |
>| other | Undefined |
>| unknown | Initializing, true ability not yet known |
>| 10BASE-T | 10BASE-T as defined in clause 14 |
>| 10BASE-TFD | Full-duplex 10BASE-T |
>| 100BASE-TX | 100BASE-TX as defined in clause 25 |
>| 100BASE-TXFD | Full-duplex 100BASE-TX |
>| 100BASE-T4 | 100BASE-T4 as defined in clause 23 |
>| isoethernet | IEEE Std 802.9 ISLAN-16T |

BEHAVIOUR DEFINED AS:
>This indicates the technology ability of the local hardware, as defined in clause 28.;

30.6.1.1.6 aAutoNegAdvertisedTechnologyAbility

ATTRIBUTE

APPROPRIATE SYNTAX:
>Same as aAutoNegLocalTechnologyAbility

BEHAVIOUR DEFINED AS:
>This GET-SET attribute maps to the Technology Ability Field of the Auto-Negotiation Link Code Word, defined in clause 28. A SET operation to a value not available in aAutoNegLocalTechnologyAbility will be rejected. A successful set operation will result in immediate link renegotiation if aAutoNegAdminState is enabled.
>
>NOTE—This will in every case cause temporary link loss during link renegotiation. If set to a value incompatible with aAutoNegReceivedTechnologyAbility, link negotiation will not be successful and will cause permanent link loss.;

30.6.1.1.7 aAutoNegReceivedTechnologyAbility

ATTRIBUTE

APPROPRIATE SYNTAX:
>Same as aAutoNegLocalTechnologyAbility

BEHAVIOUR DEFINED AS:
>Indicates the advertised technology ability of the remote hardware. Maps to the Technology Ability Field of the last received Auto-Negotiation Link Code Word(s), defined in clause 28.;

30.6.1.1.8 aAutoNegLocalSelectorAbility

ATTRIBUTE

APPROPRIATE SYNTAX:
>A SEQUENCE that meets the requirements of the description below:
>| | |
>|---|---|
>| other | Undefined |
>| ethernet | IEEE Std 802.3 |
>| isoethernet | IEEE Std 802.9 ISLAN-16T |

BEHAVIOUR DEFINED AS:
>This indicates the value of the selector field of the local hardware. Selector field is defined in 28.2.1.2.1. The enumeration of the Selector Field indicates the standard that defines the remaining encodings for Auto-Negotiation using that value of enumeration. Additional future enumerations may be assigned to this attribute through the 802.3 maintenance process.;