6. Token ring management

This clause specifies the primitives necessary to manage operation of a token ring station. This clause also defines the managed objects specific to a token ring station.

6.1 Station management primitives

This clause specifies the primitives necessary to manage the operation of the MAC and PHY as a station and also to support additional management functions. Specification of these primitives do not specify any particular implementation but rather are supplied solely to define the management of the MAC and PHY entities.

The support of the parameters, events, and actions accessible through the primitives is necessary only to support management applications. The only required primitives are the MGT_ACTION.request(acOpen) and the MGT_ACTION.request(acClose). The remaining primitives and their attributes are defined for interoperability with other standards and thus are only required as specified by other applications.

6.1.1 Token ring actions

MGT_ACTION.request { action
parameters }

MGT_ACTION.response { action
status }

The action may be acOpen or acClose, which are defined as follows:

— **acOpen:** Management instructs the MAC to join the station to the ring. This action issues the Connect.MAC event to the MAC protocol. Upon completion (success or failure) of the join ring process, an Open_status is returned to the MGT interface. The parameters passed with this action are the individual MAC address (IndMACAddress) and the ring rate (RingRate). If the MAC address is not specified, then the MAC protocol will use the MAC address assigned by the manufacturer (a universally administered). The MAC address parameter provides the means for management to set a locally administered address. If the specified address (1) is not an individual locally administered address or (2) if it is not specified and no universally administered address is assigned, then the status returned in the MGT_ACTION.response is Invalid_address. If the requested data rate is not supported, then the status returned is Rate_not_supported.

  The MGT_ACTION.response is generated when the station has successfully joined the ring or when the join process has failed. The returned status is defined as follows:

Table 8—MGT_ACTION.response definition

Open status	Table 7 REF	Definition
sNotOperational	n/a	Rejected—device not ready, internal error, failure, etc.
sInvalidAddress	n/a	Rejected—MAC address is invalid (group or not specified)
sRateNotSupported	n/a	Rejected—the requested data rate is not supported
sSuccess	100, 131, 188	Successful
sFailGeneral	n/a	Failure in activation—general
sFailLobeTest	276	Failure in activation—lobe test failure
sFailDataRate	238	Failure in activation—medium rate error
sFailDuplicate	127	Failure in activation—duplicate address
sFailDATTimeout	282	Failure in activation—DAT time-out—TJR expired
sFailNNTimeout	284	Failure in activation—neighbor notification time-out
sFailInitTimeout	286, 317	Failure in activation—initialization time-out
sFailRingBeaconing	067, 068, 069, 070, 071	Failure in activation—beacon received
sFailPurgeTimeout	318, 319, 320, 321	Failure in activation—purge failure—TRP expired
sFailClaimTimeout	265, 269, 270	Failure in activation—claim failure—TCT expired
sFailBeaconTimeout	263	Failure in activation—beacon failure—TBT expired
sFailSignalLoss	209, 210, 211	Failure in activation—signal loss
sFailInternal	224, 225, 226, 228, 229	Failure in activation—internal error
sFailTxFault	255, 256, 258, 259	Failure in activation—station transmit failure
sRemove	156, 157, 158, 160, 161	Failure in activation—remove frame received

— **acClose:** Management instructs the MAC protocol to remove the station from the ring. There are no parameters associated with this action. This action issues the Disconnect.MAC event to the MAC protocol. When the station has completed removing from the ring (i.e., transitions to bypass (JS=BP) as per table 7 REF 325), the MGT_ACTION.response is indicated with the status of Complete.

6.1.2 Token ring events

MGT_EVENT.indication { event,
event_status }

An event report is an information unit that is used to inform MGT of a significant activity in the operation of the resource. A description of the events defined for token ring follows the list of events:

evRingNonOperational
evRingOperational
evRingBeaconing
evStationFailure
evProtocolError

6.1.2.1 The **evRingNonOperational** event is generated when the monitor FSM transitions into recovery (MS=TCT, RCT, TBN, RBN). The Event_status is defined as follows:

Table 9—evRingNonOperational Event_status

Event_status	Table 7 REF	Definition
sRingNonOp	n/a	Ring not operational—cause not specified
sPurgeFailure	322	Claim—purge failure
sRingClaiming	108, 115, 116, 117, 118	Claim—claim received
sClaimNoMonitor	026, 310, 336, 337,	Claim—no active monitor (TJR, TNT, TSM=E)
sClaimSignalLoss	214, 215	Claim—signal loss
sRingBeaconing	075, 078	Beacon received

6.1.2.2 The **evRingOperational** event is generated when the ring transitions into normal repeat (MS=RPT) as per table 7 REF 169, 175, 177. The Event Status is defined as follows:

Table 10—evRingOperational Event_status

Status	Table 7 REF	Definition
sRingNonOp	175, 177, 169	Ring operational—monitor state not specified
sStandbyMonitor	175, 169	This station is a standby monitor (FAM=0)
sActiveMonitor	177	This station is the active monitor (FAM=1)

6.1.2.3 The **evRingBeaconing** event is generated when the monitor FSM transitions into a beacon state (MS=TBN, RBN) as per table 7 REF 076, 078, 073, 075, 273, 271, 272, 266. The Event_status is defined as the SA, beacon type, reported UNA, and reported physical drop number from the received beacon frame if MS=RBN or the transmitted beacon frame if MS=TBN.

6.1.2.4 The **evStationFailure** event is generated when a station fault causes the station to be removed from the ring. The Event_status is defined as follows:

Table 11—evStationFailure Event_status

Status	Table 7 REF	Definition
sRingNonOp	n/a	Station removed—cause not specified
sRemove	161	Remove frame received
sWireFault	345	Wire fault
sFailInternal	227, 257	Internal failure
sFailTest	278	Beacon test failure

6.1.2.5 The **evProtocolError** event is generated when the MAC protocol detects an error. The event status is defined as either sActiveMonitorError or sNotificationIncomplete.

Table 12—evProtocolError Event_status

Status	Table 7 REF	Definition
sActiveMonitorError	061, 104, 171, 322	Active monitor error
sNotificationIncomplete	260	Neighbor notification incomplete

The status for the active monitor error includes the error code (EC) and the status for the notification incomplete includes the address of the last station to participate (LMP).

6.1.3 Attributes

MGT_GET.request { attributeID }

MGT_REPLACE.request { attributeID,
newValue }

MGT_GET.request: The MGT_GET is issued to request information (attributes) about the token ring station. The effect of the request is that the requested attribute is returned. There is only one parameter associated with the Get request and it indicates the attributes group to be supplied in the response.

MGT_REPLACE.request: The MGT_REPLACE is issued to change certain characteristics (attributes) that effect the operation of the token ring station. The parameters indicate the attribute to change and the desired value of the attribute. The effect of the request is that the attribute is modified and a response is generated containing the value of the attribute after modification. Note that some values of attributes listed as GET-REPLACE may not be supported by the station and therefore may not change.

The attributes of a resource indicate its state (present or past) and control its operation (in the future). The Token Ring station attributes are specified in groups. Attributes are classified as follows:

a) *Characteristics.* Operational information that describes some aspect of the resource's capabilities. In general, characteristics affect the operation of the resource at some future time. Characteristics may be specifically defined to be GET (read-only) or GET-REPLACE (read-write) with respect to remote management access.

b) *Status.* Dynamic information about the resource's present state. Status attributes are read-only.

c) *Statistics.* Information about the resource's past behavior. Statistical attributes are typically a form of an event log, such as counters. The only type of statistics defined for this standard are counters that are read-only with no reset control.

6.1.3.1 Characteristics

Characteristics are the reference data of a resource that may be either necessary or useful to operate or manage the resource.

6.1.3.1.1 Station characteristics

These attributes specify the operating characteristics of the station in general. This group represents attributes that are characteristic of network operation and some of these parameters may optionally be set by management while others can not be altered by management. The characteristics of the station are

IndMACAddress—This parameter specifies the address used in the SA field of the MAC frames originated by this station. This attribute may only be set by the Open command and thus cannot be changed while the MAC is active.

UniversallyAdministeredAddress—This parameter specifies the universally administered address (if any) assigned to this station. This value can not be set by management.

FunctionalAddresses—This specifies the set of functional addresses active in the station (i.e., MAC will set the A bits and copy the frame). This parameter provides the means for management to activate functional addresses.

GroupAddresses—This parameter specifies a list of group addresses recognized by the station (i.e., MAC will set the A bits and copy the frame). This parameter provides the means for management to establish group addresses.

MicrocodeLevel—This parameter specifies the value of the X'23' Ring Station Version Number subvector (3.3.4) used in the Report Station State MAC frame. This value can not be set by management.

ProductInstanceId—This parameter specifies the value of the X'22' Product Instance ID subvector (3.3.4) used in the Report Station Attachment and Report New Active Monitor MAC frames. This value can not be set by management.

MaxFrameSize—The value of this parameter specifies the size, in octets, of the largest frame that may be transmitted and received by the station.

6.1.3.1.2 Protocol configuration options

These attributes specify the operating characteristics of the MAC protocol (refer to clause 3.5). This group represents attributes that may optionally be set by management.

Burst4Option—This parameter specifies the value of the flag FBEO indicating whether the station compensates for burst4 errors when stripping frames. The value is either Compensates or No_compensation.

BeaconHandling—This parameter specifies the value of the flag FBHO indicating whether the station waits until JS=JC before fully participating ring recovery. The value is either Before_initialization or After_initialization

ClaimContender—This parameter specifies the value of the flag FCCO indicating whether the station always actively participates in the claiming process. The value is either Contender or No_contender.

EarlyTokenRelease—This parameter specifies the value of the flag FETO indicating if early token release is selected. The value is either Enabled or Not_enabled.

RingRate—This parameter specifies the value of the station's operating speed use to set the flag FMRO. The value is either 4_Mbit/s or 16_Mbit/s. This attribute may only be set by the Open command and thus cannot be changed while the MAC is active.

MultipleFrameTransmission—This parameter specifies the value of the flag FMFTO indicating whether the station honors the reservation field when transmitting multiple frames. The value is either Relinquish or Continue.

RejectRemove—This parameter specifies the value of the FRRO indicating whether the station will honor the Remove MAC frame. The value is either Removes or Rejects.

TokenErrorDetection—This parameter specifies the value of the FTEO indicating whether the station counts errors in Tokens. The value is either Counted or Not_Counted

TokenHandling—This parameter specifies the value of the FTHO indicating how the station detects lost tokens. The value is either Active_timer or Periodic_inspection.

ErrorCountingMethod—This parameter specifies the value of the FECO indicating how the station manages the error report timer. The value is either Triggered or Free_running.

6.1.3.2 Status

A status attribute is one that indicates something about the current state of the resource. A status attribute is distinguished from a characteristic in that it is modified internally by the resource rather than by an external management entity. Status attributes are read-only.

6.1.3.2.1 Station operating status

NAUN—This parameter specifies the MAC address and physical drop number of this station's nearest upstream neighbor. The parameter values are derived from the AMP or SMP frame (table 7 REF 064, 199, 200).

RingNumber—This parameter specifies the value set by the X'03' Local Ring Number subvector (3.3.4) from the Change Parameters or Initialize Station MAC frame.

AccessPriority—This parameter specifies the value set by the X'07' Authorize Access Priority subvector (3.3.4) from the Change Parameters MAC frame.

ErrorReportTimer—This parameter specifies the value of the timer TER as set by the X'05' Error Timer Value subvector (3.3.4) from the Change Parameters or the Initialize Station MAC frame.

AuthorizeFunctionClasses—This parameter specifies the value set by the X'06' Authorize Function Classes subvector (3.3.4) of the Change Parameters MAC frame.

PhysicalDropNumber—This parameter specifies the value set by the X'04' Assign Physical Drop Number subvector (3.3.4) of the Change Parameters or the Initialize Station MAC frame.

RingStationStatus—This parameter specifies the value of the X'29' Ring Station Status subvector (3.3.4) used in the Report Station State MAC frame.

6.1.3.2.2 MAC Protocol Status

JoinState—This parameter specifies the present state of the Join FSM. The value will be one of the following: NotSpecified, Bypass, LobeTest, DetectMonitorPresent, AwaitNewMonitor, DuplicateAddressTest, NeighborNotification, RequestInitialization, JoinComplete, or BypassWait.

MonitorState—This parameter specifies the present state of the Monitor FSM. The value will be one of the following: NotSpecified, StandbyMonitor (MS=RPT & FAM=0), ActiveMonitor (MS=RPT & FAM=1), RingPurge (MS=TRP), ClaimTransmit (MS=TCT), ClaimRepeat (MS=RCT), BeaconTransmit (MS=TBN), BeaconRepeat (MS=RBN), or BeaconTest (MS=BNT).

BeaconInfo—These parameters specify the MAC address of the beaconing station, the beacon type, the MAC address of the beaconing station's upstream neighbor, and the physical drop number of the beaconing station. These parameters are the value of the SA, the X'01' Beacon Type subvector, the X'02' UNA subvector, and the X'0B' Physical Drop Number subvector in the last Beacon MAC frame transmitted or received.

TxBeaconType—This parameter specifies the value of the X'01' Beacon Type subvector in the last Beacon MAC frame transmitted.

6.1.3.3 Statistics

Statistics are attributes that contain a record of events over some period of time. The statistics defined for this standard are counters with no reset control. Access to the counterValue parameter is read-only. Refer to 3.6 for definition of each of the error counters.

6.1.3.3.1 Isolating error counters group

The isolating error counters group lists the counters and their definitions. These errors are those that can be isolated to a particular fault domain (this station, its upstream neighbor, and the wire between them). The following indicates the parameters reported and the FSM transition that causes the counter to increment:

LineErrorCounter—This counter is incremented each time CLE is incremented (table 7 REF 204, 205, 308, 309).

BurstErrorCounter—This counter is incremented each time CBE is incremented (table 7 REF 001, 002).

AcErrorCounter—This counter is incremented each time CACE is incremented (table 7 REF 194, 195, 196, 197).

AbortErrorCounter—This counter is incremented each time CABE is incremented (table 7 REF 252, 253, 302, 303).

InternalErrorCounter—This counter is incremented each time CIE is incremented (table 7 REF 222, 223).

6.1.3.3.2 Non-isolating error counters group

The non-isolating error counters group lists the counters and their definitions. These errors are those that cannot be isolated to a particular fault domain. The following indicates the parameters reported and the FSM transition that causes the counter to increment:

LostFrameErrorCounter—This counter is incremented each time CLFE is incremented (table 7 REF 031, 032).

ReceiveCongestionErrorCounter—This counter is incremented each time CRCE is incremented (table 7 REF 152, 153).

FrameCopiedErrorCounter—This counter is incremented each time CFCE is incremented (table 7 REF 041, 042).

FrequencyErrorCounter—This counter is incremented each time CFE is incremented (table 7 REF 147, 148)

TokenErrorCounter—This counter is incremented each time CTE is incremented (table 7 REF 052, 053, 291, 292, 339, 340, 342, 343).

6.2 Token ring station managed objects

This clause contains the managed objects for the token ring station. These managed objects are specified in conformance with ISO 10742 : 1994. The service and protocol elements that are involved in the management of token ring stations are defined in ISO/IEC 15802-2 : 1995.

A managed object is defined as the smallest accessible entity that contains managed attributes, actions. and event notifications that are independent and unrelated to duplicate copies of the same attributes, actions, and event notifications that exist in another managed entity.

The MAC sublayer is defined by the TokenRingLayer2MAC managed object.

6.2.1 TokenRingLayer2MAC MANAGED OBJECT CLASS

```
oTokenRingLayer2MAC MANAGED OBJECT CLASS
    DERIVED FROM "10165-2 : 1992":top;mACDLE;
    CHARACTERIZED BY
        pTokenRingMAC;
    CONDITIONAL PACKAGES
        pTrStationCharacteristics  PRESENT IF supported,
        pTokenRingMACOptions  PRESENT IF supported,
        pTrCharacteristics  PRESENT IF supported,
        pTokenRingStatus  PRESENT IF supported,
        pTokenRingStatistics  PRESENT IF supported;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) managedobjectclass(3) tokenrin-
        glayer2mac(0)};
```

6.2.2 TokenRingLayer2MAC NAME BINDING

```
nTokenRingLayer2MAC NAME BINDING
    SUBORDINATE OBJECT CLASS TokenRingLayer2MAC;
    NAMED BY
    SUPERIOR OBJECT CLASS macDLE;
    WITH ATTRIBUTE ID UniversallyAdministratedAddress
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) namebinding(6)
        tokenringlayer(0)};
```

6.2.3 TokenRingLayer2MAC PACKAGES

6.2.3.1 TokenRingMAC PACKAGE

```
pTokenRingMAC PACKAGE
    BEHAVIOR bTokenRingLayer2MAC BEHAVIOR
    DEFINED AS !This package identifies the various MAC addresses related to the station and provides
        the general management operation.!;
    ATTRIBUTES
                                aIndMACAddress   GET,          -- 0
                                aUniversalAddress GET,         -- 1
                                aFunctionalAddresses GET-REPLACE, -- 6
                                aGroupAddresses  GET-REPLACE,  -- 7
                                aNAUN            GET;          -- 22
    ATTRIBUTE GROUPS
                            "gTrStationAddresses"              -- 0
    ACTIONS
                                acOpen,                        -- 0
                                acClose;                       -- 1
    NOTIFICATIONS
                            cvStationFailure;                  -- 3
    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        tokenringmac(0)};
```

6.2.3.2 TrStationCharacteristics PACKAGE

```
pTrStationCharacteristics PACKAGE
    BEHAVIOR bTokenRingLayer2MAC BEHAVIOR
    DEFINED AS !This package identifies the characteristics of the ring station.!;
    ATTRIBUTES
                                aMicrocodeLevel  GET,          -- 2
                                aProductInstanceId GET,        -- 3
                                aMaxFrameSize    GET;          -- 4
    ATTRIBUTE GROUPS
                            "gTrStationCharacteristics"        -- 1
    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        trstationcharacteristics(1)};
```

6.2.3.3 TokenRingMACOptions PACKAGE

```
pTokenRingMACOptions PACKAGE
    BEHAVIOR  bTokenRingLayer2MAC  BEHAVIOR
    DEFINED AS !This package identifies the protocol configuration of the ring station.!;
    ATTRIBUTES
                                aBurst4Option   GET-REPLACE,      -- 8
                              aBeaconHandling   GET-REPLACE,      -- 9
                               aClaimContender  GET-REPLACE,      -- 10
                            aEarlyTokenRelease  GET-REPLACE,      -- 11
                                   aRingRate    GET,              -- 5
                     aMultipleFrameTransmission GET-REPLACE,      -- 12
                                aRejectRemove   GET-REPLACE,      -- 13
                            aTokenErrorDetection GET-REPLACE,     -- 14
                              aTokenHandling    GET-REPLACE,      -- 15
                           aErrorCountingMethod GET-REPLACE;      -- 16
    ATTRIBUTE GROUPS
                                "gTrProtocolOptions"              -- 2

    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        tokenringmacoptions(2)};
```

6.2.3.4 TrCharacteristics PACKAGE

```
pTrCharacteristics PACKAGE
    BEHAVIOR  bTokenRingLayer2MAC  BEHAVIOR
    DEFINED AS !This package identifies the ring characteristics for the station.!;
    ATTRIBUTES
                                  aRingNumber   GET{-REPLACE},    -- 17
                               aErrorReportTimer GET{-REPLACE},   -- 18
                          aAuthorizeFunctionClass GET{-REPLACE},  -- 19
                                aAccessPriority GET{-REPLACE},    -- 20
                             aPhysicalDropNumber GET{-REPLACE},   -- 21
                               aRingStationStatus GET;            -- 23
    ATTRIBUTE GROUPS
                                "gTrCharacteristics"              -- 3
    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        trcharacteristics(3)};
```

6.2.3.5 TokenRingStatus PACKAGE

```
pTokenRingStatus PACKAGE
    BEHAVIOR  bTokenRingLayer2MAC  BEHAVIOR
    DEFINED AS !This package identifies the protocol state of the ring station.!;
    ATTRIBUTES
                                    aJoinState  GET,              -- 24
                                  aMonitorState GET,              -- 25
                                   aBeaconInfo, GET,              -- 26
                                 aTxBeaconType, GET;              -- 27
```

```
    ATTRIBUTE GROUPS
                        gTrMACStatus                    -- 4

    NOTIFICATIONS
                    evRingNonOperational,               -- 0
                      evRingOperational,                -- 1
                      evRingBeaconing,                  -- 2
                      evProtocolError;                  -- 4
    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        tokenringstatus(4)};
```

6.2.3.6 TokenRingStatistics PACKAGE

```
pTokenRingStatistics  PACKAGE
    BEHAVIOR   bTokenRingLayer2MAC   BEHAVIOR
    DEFINED AS !This package provides the error statistics of the ring station.!;
    ATTRIBUTES
                        aLineErrorCounter   GET,         -- 28
                        aBurstErrorCounter  GET,         -- 29
                          aAcErrorCounter   GET,         -- 30
                     aAbortTransmitCounter  GET,         -- 31
                      aInternalErrorCounter GET,         -- 32
                    aLostFrameErrorCounter  GET,         -- 33
                aRcvCongestionErrorCounter  GET,         -- 34
                  aFrameCopiedErrorCounter  GET,         -- 35
                  aFrequencydErrorCounter   GET,         -- 36
                      aTokenErrorCounter    GET;         -- 37
    ATTRIBUTE GROUPS
                          gTrIsolatingErrors             -- 5
                        gTrNonIsolatingErrors            -- 6
    REGISTERED AS {iso(1) memberbody(2) us(840) 802dot5(10033) package(4)
        tokenringstatistics(5)};
```

6.2.4 TokenRingLayer2MAC BEHAVIOR

```
bTokenRingLayer2MAC  BEHAVIOR
    DEFINED AS !This object class defines the MAC sublayer for token ring.  The operation of the
        Token Ring MAC is defined in the ISO/IEC 8802-5 : 1995 Standard (802dot5).!;
```

6.2.5 Attributes

6.2.5.1 Individual MAC Address ATTRIBUTE

```
aIndMACAddress ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        MACDefinitions.MACAddress;
    MATCHES FOR EQUALITY;
    BEHAVIOR
        bIndMACAddress  BEHAVIOR
    DEFINED AS !Specifies the MAC address of the token ring station.!;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) attribute(7)
        indmacaddress (0)};
```

6.2.5.2 Universal Address ATTRIBUTE

```
aUniversalAddress ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        MACDefinitions.MACAddress;
    MATCHES FOR EQUALITY;
    BEHAVIOR
        bUniversalAddress BEHAVIOR
    DEFINED AS !Specifies the universal address assigned to the device.!;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) attribute(7)
        universaladdress(1)};
```

6.2.5.3 Microcode Level ATTRIBUTE

```
aMicrocodeLevel  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        MACDefinitions.MicrocodeLevel;
    MATCHES FOR EQUALITY;
    BEHAVIOR
        bMicrocodeLevel  BEHAVIOR
    DEFINED AS !Specifies the microcode level of this device.!;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) attribute(7)
        microcodelevel  (2)};
```

6.2.5.4 Product ID ATTRIBUTE

```
aProductInstanceId  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        MACDefinitions.ProductInstanceId;
    MATCHES FOR EQUALITY;
    BEHAVIOR
        bProductInstanceId  BEHAVIOR
    DEFINED AS !Identifies Product type of the Token Ring device.!;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) attribute(7)
        productinstanceid(3)};
```

6.2.5.5 Maximum Frame Size ATTRIBUTE

```
aMaxFrameSize  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        MACDefinitions.MaxFrameSize;
    MATCHES FOR EQUALITY;
    BEHAVIOR
        bMaxFrameSize BEHAVIOR
    DEFINED AS !Specifies the maximum frame size supported by the device.!;
    REGISTERED AS {iso(1) member-body(2) us(840) 802dot5(10033) attribute(7)
        maxframesize(4)};
```