

Network Working Group
Request for Comments: 1230

K. McCloghrie
Hughes LAN Systems, Inc.
R. Fox
Synoptics, Inc.
May 1991

IEEE 802.4 Token Bus MIB

Status of this Memo

This memo defines a MIB for the IEEE 802.4 Token Bus for use with the SNMP protocol. This memo is a product of the Transmission Working Group of the Internet Engineering Task Force (IETF). This RFC specifies an IAB standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Table of Contents

1. Abstract	1
2. The Network Management Framework.....	2
3. Objects	2
3.1 Format of Definitions	3
4. Overview	3
4.1 Scope of Definitions	3
4.2 Textual Conventions	4
4.3 Optional Objects	4
5. Definitions	4
6. Acknowledgements	22
7. References	22
8. Security Considerations.....	23
9. Authors' Addresses.....	23

1. Abstract

This memo defines an experimental portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, this memo defines managed objects used for managing subnetworks which use the IEEE 802.4 Token Bus technology described in 802.4 Token-Passing Bus Access Method and Physical Layer Specifications, IEEE Standard 802.4.

2. The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

RFC 1155 which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. RFC 1212 defines a more concise description mechanism, which is wholly consistent with the SMI.

RFC 1156 which defines MIB-I, the core set of managed objects for the Internet suite of protocols. RFC 1213, defines MIB-II, an evolution of MIB-I based on implementation experience and new operational requirements.

RFC 1157 which defines the SNMP, the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

3. Objects

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) [7] defined in the SMI. In particular, each object has a name, a syntax, and an encoding. The name is an object identifier, an administratively assigned name, which specifies an object type. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the OBJECT DESCRIPTOR, to also refer to the object type.

The syntax of an object type defines the abstract data structure corresponding to that object type. The ASN.1 language is used for this purpose. However, the SMI [3] purposely restricts the ASN.1 constructs which may be used. These restrictions are explicitly made for simplicity.

The encoding of an object type is simply how that object type is represented using the object type's syntax. Implicitly tied to the notion of an object type's syntax and encoding is how the object type is represented when being transmitted on the network.

The SMI specifies the use of the basic encoding rules of ASN.1 [8], subject to the additional requirements imposed by the SNMP.

3.1. Format of Definitions

Section 5 contains contains the specification of all object types contained in this MIB module. The object types are defined using the conventions defined in the SMI, as amended by the extensions specified in [9,10].

4. Overview

This memo defines three tables:

- the 802.4 Operational Table containing state and operational parameter information which is specific to 802.4 interfaces;
- the 802.4 Initialization Table containing the parameter information used by an 802.4 interface as the values to be assigned to its operational parameters upon initialization; and
- the 802.4 Statistics Table containing 802.4 interface statistics.

A managed system will have one entry in each of these tables for each of its 802.4 interfaces.

This memo also defines OBJECT IDENTIFIERS, some to identify 802.4 tests, for use with the ifExtnsTestTable defined in [11], and some to identify Token Bus interface Chip Sets, for use with the ifExtnsChipSet object defined in [11].

4.1. Scope of Definitions

All objects defined in this memo are registered in a single subtree within the experimental namespace [3], and are for use with every interface which conforms to the IEEE 802.4 Token Bus Access Method [10]. At present, this applies to interfaces for which the ifType variable in the Internet-standard MIB [4,6] has the value:

iso88024-tokenBus(8)

For these interfaces, the value of the ifSpecific variable in the MIB-II [6] has the OBJECT IDENTIFIER value:

dot4 OBJECT IDENTIFIER ::= { experimental 7 }

as defined below.

4.2. Textual Conventions

Two new datatypes, `MacAddress` and `OctetTime`, are introduced as textual conventions in this document. These textual conventions have NO effect on either the syntax nor the semantics of any managed object. Objects defined using these conventions are always encoded by means of the rules that define their primitive type. Hence, no changes to the SMI or the SNMP are necessary to accommodate these textual conventions which are adopted merely for the convenience of readers and writers in pursuit of the elusive goal of a concise and unambiguous specification.

4.3. Optional Objects

A few objects are defined in this memo with "optional" status for the purpose of allowing experimentation to determine whether they are useful or not. If sufficient consensus is reached in the Internet community to result in a subsequent revision of this memo being placed in the Internet-standard MIB, then these optional objects will either be removed or become mandatory.

5. Definitions

```
RFC1230-MIB DEFINITIONS ::= BEGIN
```

```
--                               IEEE 802.4 Token Bus MIB
```

```
IMPORTS
```

```
    experimental
        FROM RFC1155-SMI
    OBJECT-TYPE
        FROM RFC-1212;
```

```
-- This MIB Module uses the extended OBJECT-TYPE macro as
-- defined in [9].
```

```
dot4    OBJECT IDENTIFIER ::= { experimental 7 }
```

```
-- All representations of MAC addresses in this MIB Module
-- use, as a textual convention (i.e. this convention does
-- not affect their encoding), the data type:
```

```
MacAddress ::= OCTET STRING (SIZE (6))    -- a 6 octet
                                           -- address in the
                                           -- "canonical" order
```

```

-- defined by IEEE
-- 802.1a.
-- 16-bit addresses, if needed, are represented by setting
-- their upper 4 octets to all 0's, i.e., AAFF would be
-- represented as 00000000AAFF.

-- This specification follows the 802.4 convention of
-- specifying time intervals, which are dependent on the
-- bandwidth of the media, in units of octet time. One
-- octet time is the time taken to transmit eight bits.
-- Representation of such time intervals in this MIB Module
-- use, as a textual convention (i.e., this convention does
-- not affect their encoding), the data type:

OctetTime ::= INTEGER          -- the value of a time
                                -- interval in units of octet
                                -- time.

-- The 802.4 Operational Table

-- This table contains state and parameter information which
-- is specific to 802.4 interfaces. It is mandatory that
-- systems having 802.4 interfaces implement this table in
-- addition to the generic interfaces table [4,6] and its
-- generic extensions [11].

dot4Table OBJECT-TYPE
    SYNTAX  SEQUENCE OF Dot4Entry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "This table contains Token Bus interface
        parameters and state variables, one entry
        per 802.5 interface."

    ::= { dot4 1 }

dot4Entry OBJECT-TYPE
    SYNTAX  Dot4Entry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A list of Token Bus status and operational
        parameter values for an 802.4 interface."
    INDEX   { dot4IfIndex }
    ::= { dot4Table 1 }

```

```
Dot4Entry ::= SEQUENCE {
    dot4IfIndex
        INTEGER,
    dot4Options
        INTEGER,
    dot4State
        INTEGER,
    dot4Commands
        INTEGER,
    dot4MacAddrLen
        INTEGER,
    dot4NextStation
        MacAddress,
    dot4PreviousStation
        MacAddress,
    dot4SlotTime
        OctetTime,
    dot4LastTokenRotTime
        OctetTime,
    dot4HiPriTokenHoldTime
        OctetTime,
    dot4TargetRotTimeClass4
        OctetTime,
    dot4TargetRotTimeClass2
        OctetTime,
    dot4TargetRotTimeClass0
        OctetTime,
    dot4TargetRotTimeRingMaint
        OctetTime,
    dot4RingMaintTimerInitValue
        OctetTime,
    dot4MaxInterSolicitCount
        INTEGER (16..255),
    dot4MaxRetries
        INTEGER (0..7),
    dot4MinPostSilencePreambleLen
        INTEGER,
    dot4StandardRevision
        INTEGER
}
```

```
dot4IfIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The value of this object identifies the
```

802.4 interface for which this entry contains management information. The value of this object for a particular interface has the same value as the ifIndex object, defined in [4,6], for the same interface."

::= { dot4Entry 1 }

dot4Options OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The optional parts of the 802.4 specification which are in use by this station. The options of the 802.4 specification are represented by the following values:

1 - Priority

2 - Request-With-Response

The value of this object is given by the sum of the above representations for those options in use on this interface. The value zero indicates that no options are in use."

::= { dot4Entry 2 }

dot4State OBJECT-TYPE

SYNTAX INTEGER {
 other(1),
 offline(2),
 outOfRing(3),
 enteringRing(4),
 inRing(5)
}

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The current state of the 802.4 interface. The value of other(1) is used if the state is unknown (e.g., due to an error condition)."

::= { dot4Entry 3 }

dot4Commands OBJECT-TYPE

SYNTAX INTEGER {
 no-op(1),
 enterRing(2),

```

        exitRing(3),
        reset(4),
        initialize(5)
    }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
    "Setting this object causes the station
    to change the state of the interface as
    indicated by the specified value.  An
    initialize(5) command causes the
    interface to load its operational
    parameters from its initialization
    parameters; the value of
    dot4InitInRingDesired determines
    whether the station tries to enter the
    logical ring immediately.
    Note that the 802.4 specification
    suggests a station remain Offline after a
    'remote Network Management' reset(4),
    until a 'local Network Management'
    initialize(5) is performed.
    Setting this object to a value of
    no-op(1) has no effect.  When read,
    this object always has the value no-op(1)."
```

::= { dot4Entry 4 }

```

dot4MacAddrLen OBJECT-TYPE
    SYNTAX  INTEGER {
                sixteenBit(1),
                forty-eightBit(2)
            }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "This object indicates the size of MAC
        addresses interpreted by this station."
```

::= { dot4Entry 5 }

```

dot4NextStation OBJECT-TYPE
    SYNTAX  MacAddress
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The MAC address of this station's
        successor in the logical ring."
```

::= { dot4Entry 6 }

```
dot4PreviousStation OBJECT-TYPE
    SYNTAX  MacAddress
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The source MAC address of the last token
        addressed to this station."
    ::= { dot4Entry 7 }

dot4SlotTime OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The maximum time any station need wait
        for an immediate MAC-level response from
        another station.
        This value must be the same in all stations on
        the 802.4 network."
    ::= { dot4Entry 8 }

dot4LastTokenRotTime OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The observed token rotation time for the
        last token rotation, timed from token
        arrival to token arrival.  A value of
        zero indicates that the token is not
        rotating."
    ::= { dot4Entry 9 }

dot4HiPriTokenHoldTime OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The maximum duration for which a station
        can hold the token to transmit frames of
        access class 6 (if the priority option is
        implemented), or of any access class (if
        the priority option is not implemented)."
    ::= { dot4Entry 10 }

dot4TargetRotTimeClass4 OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-only
```

STATUS mandatory

DESCRIPTION

"If the priority scheme is being used, this value specifies a limit on how long a station can transmit frames at access class 4. The limit is measured from the time the station is able to start transmitting frames at this access class on one rotation, to the time it must stop transmitting frames at this access class on the next rotation. If the priority scheme is not being used, this object has the value 0."

::= { dot4Entry 11 }

dot4TargetRotTimeClass2 OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-only

STATUS mandatory

DESCRIPTION

"If the priority scheme is being used, this value specifies a limit on how long a station can transmit frames at access class 2. The limit is measured from the time the station is able to start transmitting frames at this access class on one rotation, to the time it must stop transmitting frames at this access class on the next rotation. If the priority scheme is not being used, this object has the value 0."

::= { dot4Entry 12 }

dot4TargetRotTimeClass0 OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-only

STATUS mandatory

DESCRIPTION

"If the priority scheme is being used, this value specifies a limit on how long a station can transmit frames at access class 0. The limit is measured from the time the station is able to start transmitting frames at this access class on one rotation, to the time it must stop transmitting frames at this access class on the next rotation. If the priority scheme is not being used,

this object has the value 0."
 ::= { dot4Entry 13 }

dot4TargetRotTimeRingMaint OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-only

STATUS mandatory

DESCRIPTION

"A value used to limit the duration of a token rotation. If the duration of a token rotation exceeds this value, the station will not open the response window to solicit for a new successor."

::= { dot4Entry 14 }

dot4RingMaintTimerInitValue OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The value to which the dot4TargetRotTimeRingMaint is set, each time the station enters the ring. A large value will cause the station to solicit successors immediately upon entry to the ring; a value of zero will cause the station to defer this solicitation for at least one token rotation."

::= { dot4Entry 15 }

dot4MaxInterSolicitCount OBJECT-TYPE

SYNTAX INTEGER (16..255)

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The maximum number of consecutive token rotations without soliciting for a successor. If this count expires, the station opens the response window to solicit for a successor (providing the duration of the current token rotation has not exceeded dot4TargetRotTimeRingMaint). The least significant two bits of the count are determined randomly by the station on a per-use basis."

::= { dot4Entry 16 }

```

dot4MaxRetries OBJECT-TYPE
    SYNTAX  INTEGER (0..7)
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The maximum number of retries of a
        Request-with-Response (RWR) frame.  If
        the RWR option is not in use, this object
        has the value 0."
    ::= { dot4Entry 17 }

dot4MinPostSilencePreambleLen OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The minimum number of octets of preamble
        on the first frame transmitted by this
        station after a period of 'transmitted'
        silence."
    ::= { dot4Entry 18 }

dot4StandardRevision OBJECT-TYPE
    SYNTAX  INTEGER {
        rev2(2)
    }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The Revision number of the 802.4 standard
        implemented by this station."
    ::= { dot4Entry 19 }

```

-- 802.4 Initialization Table

```

-- This table contains the parameter information used by an
-- 802.4 interface as the values to be assigned to its
-- operational parameters upon initialization.  It is
-- mandatory that systems having 802.4 interfaces implement
-- this table.

```

```

dot4InitTable OBJECT-TYPE
    SYNTAX  SEQUENCE OF Dot4InitEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "This table contains Token Bus

```

initialization parameters, one entry per
802.4 interface."
 ::= { dot4 2 }

dot4InitEntry OBJECT-TYPE
SYNTAX Dot4InitEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
 "A list of Token Bus initialization
 parameters for an 802.4 interface."
INDEX { dot4InitIfIndex }
 ::= { dot4InitTable 1 }

Dot4InitEntry ::= SEQUENCE {
 dot4InitIfIndex
 INTEGER,
 dot4InitSlotTime
 OctetTime,
 dot4InitMaxInterSolicitCount
 INTEGER (16..255),
 dot4InitMaxRetries
 INTEGER (0..7),
 dot4InitHiPriTokenHoldTime
 OctetTime,
 dot4InitTargetRotTimeClass4
 OctetTime,
 dot4InitTargetRotTimeClass2
 OctetTime,
 dot4InitTargetRotTimeClass0
 OctetTime,
 dot4InitTargetRotTimeRingMaint
 OctetTime,
 dot4InitRingMaintTimerInitValue
 OctetTime,
 dot4InitMinPostSilencePreambleLen
 INTEGER,
 dot4InitInRingDesired
 INTEGER
}

dot4InitIfIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The value of this object identifies the
 802.4 interface for which this entry

```
        contains management information.  The
        value of this object for a particular
        interface has the same value as the
        ifIndex object, defined in [4,6], for
        the same interface."
 ::= { dot4InitEntry 1 }

dot4InitSlotTime OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "The value assigned to the object
        dot4SlotTime when the station is
        initialized."
 ::= { dot4InitEntry 2 }

dot4InitMaxInterSolicitCount OBJECT-TYPE
    SYNTAX  INTEGER (16..255)
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "The value assigned to the object
        dot4MaxInterSolicitCount when the station
        is initialized."
 ::= { dot4InitEntry 3 }

dot4InitMaxRetries OBJECT-TYPE
    SYNTAX  INTEGER (0..7)
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "The value assigned to the object
        dot4MaxRetries when the station is
        initialized."
 ::= { dot4InitEntry 4 }

dot4InitHiPriTokenHoldTime OBJECT-TYPE
    SYNTAX  OctetTime
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "The value assigned to the object
        dot4HiPriTokenHoldTime when the station
        is initialized."
 ::= { dot4InitEntry 5 }
```

dot4InitTargetRotTimeClass4 OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The value assigned to the object
dot4TargetRotTimeClass4 when the station
is initialized."

::= { dot4InitEntry 6 }

dot4InitTargetRotTimeClass2 OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The value assigned to the object
dot4TargetRotTimeClass2 when the station
is initialized."

::= { dot4InitEntry 7 }

dot4InitTargetRotTimeClass0 OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The value assigned to the object
dot4TargetRotTimeClass0 when the station
is initialized."

::= { dot4InitEntry 8 }

dot4InitTargetRotTimeRingMaint OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The value assigned to the object
dot4TargetRotTimeRingMaint when the station
is initialized."

::= { dot4InitEntry 9 }

dot4InitRingMaintTimerInitValue OBJECT-TYPE

SYNTAX OctetTime

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The value assigned to the object
dot4RingMaintTimerInitValue when the
station is initialized."

```
 ::= { dot4InitEntry 10 }
```

```
dot4InitMinPostSilencePreambleLen OBJECT-TYPE
```

```
SYNTAX INTEGER
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
    "The value assigned to the object
    dot4MinPostSilencePreambleLen when the
    station is initialized."
```

```
 ::= { dot4InitEntry 11 }
```

```
dot4InitInRingDesired OBJECT-TYPE
```

```
SYNTAX INTEGER {
```

```
    inRing(1),
```

```
    outOfRing(2)
```

```
}
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
    "This object determines whether the
    station will attempt to enter the logical
    ring immediately after initialization."
```

```
 ::= { dot4InitEntry 12 }
```

```
-- 802.4 Statistics Table
```

```
-- This table contains Token Bus statistics, one entry per
```

```
-- 802.4 interface. It is mandatory that systems having
```

```
-- 802.4 interfaces implement this table.
```

```
dot4StatsTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF Dot4StatsEntry
```

```
ACCESS not-accessible
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
    "A table containing Token Bus statistics.
    All the statistics are defined using the
    syntax Counter as 32 bit wrap around
    counters. Thus, if an interface's
    hardware chip set maintains these
    statistics in 16-bit counters, then the
    agent must read the hardware's counters
    frequently enough to prevent loss of
    significance, in order to maintain
    a 32-bit counter in software."
```

```
 ::= { dot4 3 }
```

```
dot4StatsEntry OBJECT-TYPE
    SYNTAX  Dot4StatsEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "An entry containing the 802.4 statistics
         for a particular interface."
    INDEX   { dot4StatsIfIndex }
    ::= { dot4StatsTable 1 }
```

```
Dot4StatsEntry ::= SEQUENCE {
    dot4StatsIfIndex
        INTEGER,
    dot4StatsTokenPasses
        Counter,
    dot4StatsTokenHeards
        Counter,
    dot4StatsNoSuccessors
        Counter,
    dot4StatsWhoFollows
        Counter,
    dot4StatsTokenPassFails
        Counter,
    dot4StatsNonSilences
        Counter,
    dot4StatsFcsErrors
        Counter,
    dot4StatsEbitErrors
        Counter,
    dot4StatsFrameFrag
        Counter,
    dot4StatsFrameTooLongs
        Counter,
    dot4StatsOverRuns
        Counter,
    dot4StatsDupAddresses
        Counter
}
```

```
dot4StatsIfIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The value of this object identifies the
         802.4 interface for which this entry
         contains management information. The
         value of this object for a particular
```

```
        interface has the same value as the
        ifIndex object, defined in [4,6], for the
        same interface."
 ::= { dot4StatsEntry 1 }

dot4StatsTokenPasses OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS optional
    DESCRIPTION
        "The number of times this station has
        passed the token."
 ::= { dot4StatsEntry 2 }

dot4StatsTokenHeards OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS optional
    DESCRIPTION
        "The number of tokens heard by this
        station."
 ::= { dot4StatsEntry 3 }

dot4StatsNoSuccessors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of times the station could
        not find a successor while believing
        itself not the only station in the ring.
        This can signify a faulty transmitter
        condition in this station."
 ::= { dot4StatsEntry 4 }

dot4StatsWhoFollows OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of times the station has had
        to look for a new next station."
 ::= { dot4StatsEntry 5 }

dot4StatsTokenPassFails OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
```

DESCRIPTION

"The number of times the station failed in passing the token to the next station."
 ::= { dot4StatsEntry 6 }

dot4StatsNonSilences OBJECT-TYPE

SYNTAX Counter
ACCESS read-only
STATUS mandatory

DESCRIPTION

"The number of occurrences of non-silence followed by silence in which a start delimiter was not detected."
 ::= { dot4StatsEntry 7 }

dot4StatsFcsErrors OBJECT-TYPE

SYNTAX Counter
ACCESS read-only
STATUS mandatory

DESCRIPTION

"The number of frames received with an incorrect FCS and the E-bit reset."
 ::= { dot4StatsEntry 8 }

dot4StatsEbitErrors OBJECT-TYPE

SYNTAX Counter
ACCESS read-only
STATUS mandatory

DESCRIPTION

"The number of frames the station received with the E-bit set in the end delimiter."
 ::= { dot4StatsEntry 9 }

dot4StatsFrameFrgs OBJECT-TYPE

SYNTAX Counter
ACCESS read-only
STATUS mandatory

DESCRIPTION

"The number of occurrences of receiving a start delimiter followed by another start delimiter, an invalid symbol sequence or silence, without an intervening end delimiter."
 ::= { dot4StatsEntry 10 }

dot4StatsFrameTooLongs OBJECT-TYPE

SYNTAX Counter

```

ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "The number of frames that were received
    that were larger than the media's MTU."
 ::= { dot4StatsEntry 11 }

```

```

dot4StatsOverRuns OBJECT-TYPE
    SYNTAX    Counter
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "The number of times a FIFO overrun was
        detected in the station."
    ::= { dot4StatsEntry 12 }

```

```

dot4StatsDupAddresses OBJECT-TYPE
    SYNTAX    Counter
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "The number of times this station
        detected another station using the same
        MAC address."
    ::= { dot4StatsEntry 13 }

```

```
--                               802.4 Interface Tests
```

```
dot4Tests                OBJECT IDENTIFIER ::= { dot4 5 }
```

```
-- The extensions to the interfaces table proposed in [11]
-- define a table object, ifExtnsTestTable, through which a
-- network manager can instruct an agent to test an interface
-- for various faults. A test to be performed is identified
-- (as the value of ifExtnsTestType) via an OBJECT IDENTIFIER.
```

```
-- When a test fails, the object ifExtnsTestCode, defined in
-- [11], may contain a media-specific error-code. For 802.4
-- interfaces, the following is defined as the value of
-- ifExtnsTestCode when a test fails because the modem could
-- not be initialized:
```

```
dot4Errors OBJECT IDENTIFIER ::= { dot4 4 }
dot4ModemInitFailed OBJECT IDENTIFIER ::= { dot4Errors 1 }
```

```
-- The Full-Duplex Loop Back Test is a common test, defined
-- in [11] as:
```

```
--
--      testFullDuplexLoopBack
--
-- Invoking this test on a 802.4 interface causes the
-- interface to check the path from memory through the chip
-- set's serial interface back to memory, thus checking the
-- proper functioning of the transmit and receive machines
-- of the token bus hardware.
-- This test may only be invoked when the interface is
-- the Offline state.

-- The FIFO Path test is defined by:

testFifoPath    OBJECT IDENTIFIER ::= { dot4Tests 1 }

-- Invoking this test causes the interface to check the path
-- from memory to the chip set's FIFO and back to memory.
-- This test checks the hosts interface to the token bus
-- hardware. This test may only be invoked when the
-- interface is the Offline state.

-- The External Loopback test is defined by:

testExternalLoopback OBJECT IDENTIFIER ::= { dot4Tests 2 }

-- Invoking this test causes the interface to check the path
-- from memory through the chip set and out onto the network
-- for external (e.g., at the head-end) loopback through the
-- chip set to memory. This test checks the host's interface
-- to the 802.4 network. This test is liable to cause
-- serious disruption if invoked on an operational network.

--
--      802.4 Hardware Chip Sets

dot4ChipSets    OBJECT IDENTIFIER ::= { dot4 6 }

-- The extensions to the interfaces table proposed in [11]
-- define an object, ifExtnsChipSet, with the syntax of
-- OBJECT IDENTIFIER, to identify the hardware chip set in
-- use by an interface. That definition specifies just
-- one applicable object identifier:
--
--      unknownChipSet
--
-- for use as the value of ifExtnsChipSet when the specific
-- chip set is unknown.
```

```
--
-- This MIB defines the following for use as values of
-- ifExtnsChipSet:
-- for use as values of ifExtnsChipSet

-- Motorola 68824 Token Bus Controller
chipSetMc68824 OBJECT IDENTIFIER ::= { dot4ChipSets 1 }

END
```

6. Acknowledgements

This document was produced under the auspices of the IETF's Transmission Working Group. The comments of the following individuals are acknowledged:

Brian Kline, Hughes LAN Systems, Inc.
Bruce Lieberman, Hughes LAN Systems, Inc.
Marshall T. Rose, Performance Systems International, Inc.

7. References

- [1] Cerf, V., "IAB Recommendations for the Development of Internet Network Management Standards", RFC 1052, NRI, April 1988.
- [2] Cerf, V., "Report of the Second Ad Hoc Network Management Review Group", RFC 1109, NRI, August 1989.
- [3] Rose M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based internets", RFC 1155, Performance Systems International, Hughes LAN Systems, May 1990.
- [4] McCloghrie K., and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets", RFC 1156, Hughes LAN Systems, Performance Systems International, May 1990.
- [5] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", RFC 1157, SNMP Research, Performance Systems International, Performance Systems International, MIT Laboratory for Computer Science, May 1990.
- [6] McCloghrie K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets", RFC 1213, Performance Systems International, March 1991.
- [7] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization, International

Standard 8824, December 1987.

- [8] Information processing systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Notation One (ASN.1), International Organization for Standardization, International Standard 8825, December 1987.
- [9] Rose, M., and K. McCloghrie, Editors, "Concise MIB Definitions", RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.
- [10] Token-Passing Bus Access Method and Physical Layer Specifications, Institute of Electrical and Electronic Engineers, IEEE Standard 802.4, May 1988.
- [11] McCloghrie, K., Editor, "Extensions to the Generic-Interface MIB", RFC 1229, Hughes LAN Systems, May 1991.

8. Security Considerations

Security issues are not discussed in this memo.

9. Authors' Addresses

Keith McCloghrie
Hughes LAN Systems, Inc.
1225 Charleston Road
Mountain View, CA 94043

Phone: (415) 966-7934
EMail: kzm@hls.com

Richard Fox
Synoptics, Inc.
4401 Great America Pkwy
PO Box 58185
Santa Clara, Cal. 95052

Phone: (408) 764-1372
EMail: rfox@synoptics.com