Network Working Group
Internet-Draft
Expires: April 19, 2004

E. Beili
Actelis Networks
October 20, 2003

Ethernet in the First Mile Copper (EFMCu) Interfaces MIB
draft-beili-hubmib-efm-cu-mib-00.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on April 19, 2004.

Abstract

This document defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP based Internets. This document proposes an extension to the Ethernet-like Interfaces MIB and MAU MIB with a set of objects for managing an Ethernet in the First Mile Copper (EFMCu) interfaces 10PassTS and 2BaseTL defined in IEEE 802.3ah.

Table of Contents

1. Introduction

New Ethernet like interfaces have been defined in the IEEE 802.3ah project Ethernet in the First Mile (EFM). In particular 2BaseTL and 10PassTS interfaces defined over voice-grade copper pairs have been specified. These interfaces, collectively called EFMCu, support variable rates and optional PMI aggregation (multi-pair bonding).

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community to manage EFMCu interfaces.

2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 .

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

3. Relation to Interfaces MIB

This section specifies how the ifStackTable, as defined in [RFC2863] and ifInvStackTable, as defined in [RFC2864] are used for the EFMCu application.

3.1 Layering Model

An EFMCu interface can aggregate up to 32 PMDs (modems) using so called PMI Aggregation Function (PAF).

An generic EFMCu device can have a number of MII/PCS ports cross-connected to a number of PMDs, with a single PCS per PMD relationship.

Each PMD comprising an aggregated EFMCu port is represented in the Interface table as a separate port with ifType of shdsl (169) for 2BaseTL or vdsl(97) for 10PassTS. The ifType values are defined in IANAifType-MIB. ifSpeed for each PMD shall return an actual bitrate of the active PMD or a configured bitrate for pre-activated modems (note that unassigned PMD has its default bitrate).

The ifStackTable is indexed by the ifIndex values of the aggregated EFMCu port (PCS) and the PMDs connected to it. ifStackTable allows a Network Management application to determine which PMDs are connected to a particular PCS and change connections. The ifInvStackTable, being an inverted version of the ifStackTable, provides an efficient means for a Network Management application to read a subset of the ifStackTable and thereby determine which PCS runs on top of a particular PMD.

A new table efmCuAvailableStackTable defined in this MIB, specifies for each PCS a list of PMDs, which can possibly be cross-connected to that PCS, determined by the cross-connect capability of the device. This table, modeled after ifStackTable, is read only.

Editor's Note: An alternative would be to use ifStackTable to describe cross-connect capability and efmCuAvailableStackTable to describe actual connections, so that the cross-connect action would be done in the EFM-CU-MIB by modifying the efmCuAvailableStackTable (and not in IF-MIB).

3.2 PMI Aggregation Function (PAF)

aPAFSupported is mandatory for all EFMCu ports (2BASE-TL and 10PASS-TS).

3.3 Discovery Operation

This MIB allows a Network Management application to control EFM Discovery mechanism and query its results. Note that the Discovery mechanism can work only if PAF is supported and enabled.

Two tables are used by Discovery mechanism: ifStackTable and efmCuAvailableStackTable defined. The following pseudo-code defines an example of Discovery for a generic PAF enabled multi-PCS EFMCu device, located at Central Office (CO):

```
foreach PCS[i] in Device
{ if ( PCS[i].PAFSupported ) // Discover only on ports supporting  PAF
    { dc = PCS[i].DiscoveryCode = MAC[i]; // unique 6 byte code per PCS
      // go over all currently disconnected PMDs, which can
   // pottentially be connected to PCS[i]
      foreach PMD[j] in efmCuAvailableStackTable[PCS[i]] and
                    not in ifStackTable[PCS[i]]
        { PMD[j].RemoteDiscoveryCode = dc; // Set if Clear
          r = PMD[j].RemoteDiscoveryCode;  // Get
          if ( r == dc )
            { // Remote CPE connected via PMD[j] is/was a peer for
           // PCS[i]. Connect this PMD to the PCS
              Add PMD[j] to ifStackTable[PCS[i]];
              // Discover all other currently disconnected PMDs,
              // attached to the same CPE and connect them to the PCS
              foreach PMD[k] in efmCuAvailableStackTable[PCS[i]] and
                              not in ifStackTable[PCS[i]]
                { r = PMD[k].RemoteDiscoveryCode;  // Get
                  if ( r == dc )
                    Add PMD[k] to ifStackTable[PCS[i]];
                }
            }
          // Discovered all PMDs which lead to the same CPE and
          // connected them to PCS[i]. Go to the next PCS.
          break;
        }
    }
}
```

The SNMP Agent builds efmCuStackTable according to the information contained in the Clause 45 PMI_Available_register (see [802.3ah] 61.1.5.3 and 45.2.3.20).

Adding a PMD to the ifStackTable row for a specific PCS involve actual connection of the PCS and PMD, which can be done by modifying Clause 45 PMI_Aggregate_register (see [802.3ah] 61.1.5.3 and 45.2.3.21).

3.4 Relation to SHDSL MIB

PMD(s) comprising a 2BaseTL port are described in HDSL2-SHDSL-LINE-MIB . Note that HDSL2-SHDSL-LINE-MIB describes standard G.SHDSL modems according to ITU-T G.991.2, while IEEE 802.3ah uses so called G.SHDSL.bis spec, extended to support higher constellations and rates. In cases where G.SHDSL and 802.3ah differ, the definitions in 802.3ah take precedence.

3.5 Relation to VDSL MIB

PMD(s) comprising a 10PassTS port are described in VDSL-LINE-MIB [draft-ietf-adslmib-vdsl]. In cases where VDSL-LINE-MIB and 802.3ah differ, the definitions in 802.3ah take precedence

3.6 Relation to Ethernet-Like and MAU MIBs

EFMCu interfaces require implementation of ETHERIF-MIB [draft-ietf-hubmib-etherif-mib] and MAU-MIB [RFC3636]. As such EFMCu interfaces 2BaseTL/10PassTS shall return an ifType of ethernetCsmacd(6). Information on the particular flavor of EFMCu that an interface is running is available from ifSpeed in the Interfaces Group MIB [RFC2863], and ifMauType in the MAU MIB [RFC3636].

3.7 Mapping of IEEE 802.3ah Managed Objects

This section contains the mapping between oMAU managed objects defined in [802.3ah] and managed objects defined in this document and in associated MIB modules, i.e., the IF-MIB [RFC2863], the HDSL2-SHDSL-LINE-MIB , and the MAU-MIB [RFC3636].

IEEE 802.3 Managed Object Corresponding SNMP Object

4. Definitions

```
  EFM-CU-MIB DEFINITIONS ::= BEGIN

   IMPORTS
       MODULE-IDENTITY, OBJECT-TYPE,
       Gauge32, Integer32, transmission
           FROM SNMPv2-SMI
       TruthValue, RowStatus, PhysAddress
           FROM SNMPv2-TC
       ifIndex, InterfaceIndexOrZero
           FROM IF-MIB
       MODULE-COMPLIANCE, OBJECT-GROUP
           FROM SNMPv2-CONF
       ;

   efmCuMIB MODULE-IDENTITY
       LAST-UPDATED "200310200000Z"  -- October 20, 2003
           ORGANIZATION "IETF Ethernet Interfaces and Hub MIB
                         Working Group"
           CONTACT-INFO
              "WG charter:
                 http://www.ietf.org/html.charters/hubmib-charter.html
```

Mailing Lists:
General Discussion: hubmib@ietf.org
To Subscribe: hubmib-request@ietf.org
In Body: subscribe your_email_address

Chair: Dan Romascanu
Postal: Avaya Inc.
Atidim Technology Park, Bldg. 3
Tel Aviv 61131
Israel
Tel: +972 3 645 8414
E-mail: dromasca@avaya.com

Editor: Edward Beili
Postal: Actelis Networks Inc.
25 Bazel St., P.O.B. 10173
Petach-Tikva 10173
Israel
Tel: +972-3-924-3491
E-mail: edward.beili@actelis.com"

DESCRIPTION
"The objects in this MIB module are used to manage
the Ethernet in the First Mile (EFM) Copper (EFMCu) Interfaces
2BASE-TL and 10PASS-TS, defined in IEEE Draft P802.3ah/D2.1.

The following reference is used throughout this MIB module:

[802.3ah] refers to:
IEEE Draft P802.3ah/D2.1: 'Draft amendment to -
Information technology - Telecommunications and
information exchange between systems - Local and
metropolitan area networks - Specific requirements -
Part 3: Carrier sense multiple access with collision
detection (CSMA/CD) access method and physical layer
specifications - Media Access Control Parameters, Physical
Layers and Management Parameters for subscriber access
networks', 07 October 2003.

Of particular interest are Clause 61, 'Physical Coding
Sublayer (PCS) and common specifications, type 10PASS-TS and
type 2BASE-TL', Clause 30, 'Management', and Clause 45,
'Management Data Input/Output (MDIO) Interface'.

```
        -- Editor's Note: Replace XXXX with the actual RFC number
        -- assigned by RFC Editor and remove this note

        REVISION    "200310200000Z"  -- October 20, 2003
        DESCRIPTION "Initial version, published as RFC XXXX."

        ::= { transmission 135 }

        -- Editor's Note: Replace 135 with a real OID once it is
        -- assigned by IANA and remove this note.
        -- This OID is temporary so that compilation does not to fail.

    -- Sections of the module

    efmCuObjects     OBJECT IDENTIFIER ::= { efmCuMIB 1 }

    efmCuConformance OBJECT IDENTIFIER ::= { efmCuMIB 2 }

    -- Groups in the module

    efmCuPort        OBJECT IDENTIFIER ::= { efmCuObjects 1 }

    efmCuPmd      OBJECT IDENTIFIER ::= { efmCuObjects 2 }

    -- The PCS Port group

    efmCuPortTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF EfmCuPortEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
           "Table for EFMCu 2BaseTL/10PassTS (PCS) Ports."
         ::= { efmCuPort 1 }

    efmCuPortEntry OBJECT-TYPE
        SYNTAX  EfmCuPortEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
           "An entry in the EFMCu Port table."
        INDEX  { ifIndex }
         ::= { efmCuPortTable 1 }

    EfmCuPortEntry ::=
        SEQUENCE {
            efmCuPortSidesSupported          INTEGER,
            efmCuPortSide                         INTEGER,
            efmCuPAFSupported                     TruthValue,
```

```
            efmCuPAFAdminState                  INTEGER,
         efmCuPAFDiscoveryCode            PhysAddress
            }

      efmCuPortSidesSupported  OBJECT-TYPE
          SYNTAX  INTEGER {
                   subscriber(1),
                   office(2),
                 both(3)
                }
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
             "EFM port supported mode of operation (subtype).
             The value of 'subscriber' indicates that the port supports
             'CPE' or '-R' subtype.
             The value of 'office' indicates that the port supports
             'CO' or '-O' subtype.
             The value of 'both' indicates that the port supports both
             'CO' and 'CPE' subtypes.
             An actual mode of operation is determined by ifPhySide.

             If a Clause 45 MDIO Interface to the PCS is present, then this
             attribute will map to the CO supported and CPE supported
             bits in the 10P/2B capability register"
          REFERENCE
             "[802.3ah] 61.1, 45.2.3.18.2, 45.2.3.18.3"
          ::= { efmCuPortEntry 1 }

      efmCuPortSide  OBJECT-TYPE
          SYNTAX  INTEGER {
                   subscriber(1),
                   office(2)
                }
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "EFM port mode of operation (subtype).
             The value of 'subscriber' indicates the port is designated as
             the 'CPE' or '-R' subtype.
             The value of the 'office' indicates that the port is
          designated as the 'CO' or '-O' subtype.
             Attempts to change this object to an unsupported subtype shall
          be ignored.

             If a Clause 45 MDIO Interface to the PCS is present, then this
             attribute will map to the Port sub-type select bit in the
          10P/2B capability register"
```

```
          REFERENCE
             "[802.3ah] 61.1, 45.2.3.18.1"
          ::= { efmCuPortEntry 2 }

      efmCuPAFSupported  OBJECT-TYPE
          SYNTAX      TruthValue
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
             "PMI Aggregation Function (PAF) Capability of the EFMCu port
          (PCS).
             This object has a value of true(1) when the PCS can perform
          PMI aggregation on the available PMDs.
          Ports incapable of PAF shall return a value of false(2).

             If a Clause 45 MDIO Interface to the PCS is present,
             then this attribute will map to the PAF supported bit in the
             10P/2B capability register."
          REFERENCE
             "[802.3ah] 61.2.2, 45.2.3.18.4"
          ::= { efmCuPortEntry 3 }

      efmCuPAFAdminState  OBJECT-TYPE
          SYNTAX  INTEGER {
                   enabled(1),
                   disabled(2)
               }
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "Administrative (desired) state of the PAF of the EFMCu port
          (PCS).
             When 'disabled', PMI Aggregation will not be performed by the
          PCS.
             When 'enabled', PAF will be performed by the PCS when the link
          is Up, even on a single PMD, if PAF is supported.
          PCS ports incapable of supporting PAF shall return a value of
             'disabled'. Attempts to 'enable' such port shall be ignored.

             Changing PAFAdminState is a traffic disruptive operation and
          as such shall be done when the link is Down. Attempts to
          change this object shall be ignored if the link is Up or
          Initializing.

             If a Clause 45 MDIO Interface to the PCS is present, then this
             attribute will map to the PAF enable bit in the 10P/2B
          capability register"
          REFERENCE
```

```
            "[802.3ah] 61.2.2, 45.2.3.18.4"
        ::= { efmCuPortEntry 4 }

    efmCuPAFDiscoveryCode  OBJECT-TYPE
        SYNTAX      PhysAddress
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "PAF Discovery Code of the EFMCu port (PCS).
            A unique 6 Byte long code used by the Discovery function.
            This object must be instantiated for the CO subtype PCS before
        writing operations on the PAFRemoteDiscoveryCode
            (Set_if_Clear and Clear_if_Same) are performed by PMDs
        associated with the PCS.
        The value of this object is read-only for CPE port subtypes.
        (The initial value of this object for CPE ports after reset
        is 0).

        Discovery must be performed when the link is Down.
            Attempts to change this object MUST be rejected with the error
            inconsistentValue if the link is Up or Initializing.

            If a Clause 45 MDIO Interface to the PCS is present, then this
            attribute will map to the Aggregaion Discovery Code registers"
        REFERENCE
            "[802.3ah] 61.2.2.8.3, 45.2.1.12, 45.2.1.13"
        ::= { efmCuPortEntry 5 }


    -- The PMD group

    efmCuPmdTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF EfmCuPmdEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
            "Table for EFMCu 2BaseTL/10PassTS PMDs (modems). Common part"
         ::= { efmCuPmd 1 }

    efmCuPmdEntry OBJECT-TYPE
        SYNTAX  EfmCuPmdEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
            "An entry in the EFMCu PMD Common table."
        INDEX  { ifIndex }
         ::= { efmCuPmdTable 1 }
```

```
EfmCuPmdEntry ::=
    SEQUENCE {
        efmCuPAFRemoteDiscoveryCode      PhysAddress,
        efmCuPmdRxSnrMgn              Integer32,
        efmCuPmdRemoteRxSnrMgn           Integer32
        }

efmCuPAFRemoteDiscoveryCode  OBJECT-TYPE
    SYNTAX      PhysAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
       "PAF Remote Discovery Code of the PMD port at CO.
       A 6 Byte long Discovery Code of the peer PCS connected via
    the PMD.
       Reading this object results in a Discovery Get operation.
       Writing a zero to this object results in a Discovery
    Clear_if_Same operation (the value of the ifPAFDiscoveryCode
    at the peer PCS shall be the same as ifPAFDiscoveryCode of
    the local PCS associated with the PMD for the operation to
    succeed).
       Writing a non-zero value to this object reslults in a
    Discovery Set_if_Clear operation.
    This object does not exist in CPE port subtypes. A zero length
    octet string shall be returned for CPE port subtypes and also
    when PAF aggregation is not enabled.

    Discovery must be performed when the link is Down.
       Attempts to change this object MUST be rejected with the error
       inconsistentValue, if the link is Up or Initializing.

       If a Clause 45 MDIO Interface to the PCS is present, then this
       attribute is a function of Aggregation Discovery Operation,
    Code and Operation result registers"
    REFERENCE
       "[802.3ah] 61.2.2.8.3, 45.2.1.12.1"
    ::= { efmCuPmdEntry 1 }

efmCuPmdRxSnrMgn OBJECT-TYPE
    SYNTAX      Integer32(-127..128)
    UNITS       "dB"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
    "The current Signal-to-Noise Ratio (SNR) margin with respect
    to the received signal, for the PMD.

       If a Clause 45 MDIO Interface is present, then this
```

```
            attribute will map to the Rx SNR Margin register"
        REFERENCE
            "[802.3ah] 45.2.1.17"
        ::= { efmCuPmdEntry 2 }

    efmCuPmdRemoteRxSnrMgn OBJECT-TYPE
        SYNTAX      Integer32(-127..128)
        UNITS       "dB"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
        "The current SNR margin with respect
        to the received signal, for the CO PMD's link partner
        (CPE PMD).

        This object is not supported by CPE port subtypes.

            If a Clause 45 MDIO Interface is present, then this
            attribute will map to the Remote Rx SNR Margin register"
        REFERENCE
            "[802.3ah] 45.2.1.17"
        ::= { efmCuPmdEntry 3 }

    -- 2BaseTL specific PMD group

    efmCuPmd2BTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF EfmCuPmd2BEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
            "Table for EFMCu 2BaseTL PMDs (modems)."
         ::= { efmCuPmd 2 }

    efmCuPmd2BEntry OBJECT-TYPE
        SYNTAX  EfmCuPmd2BEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
            "An entry in the EFMCu 2BaseTL PMD table."
        AUGMENTS { efmCuPmdEntry }
         ::= { efmCuPmd2BTable 1 }

    EfmCuPmd2BEntry ::=
        SEQUENCE {
            efmCuPmd2BRegion                INTEGER,
            efmCuPmd2BPower                      Integer32,
            efmCuPmd2BDataRate              Integer32,
            efmCuPmd2BConstellation         INTEGER,
```

```
              efmCuPmd2BLoopAtnThreshold       Integer32,
              efmCuPmd2BSnrMgnThreshold        Integer32
              }

      efmCuPmd2BRegion  OBJECT-TYPE
          SYNTAX  INTEGER {
                   annexA(1), -- region 1
                   annexB(2), -- region 2
                   annexC(3)  -- region 3
               }
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "Desired Power Spectral Density (PSD) Regional setting as
specified
          in Regional Annex of [ITU-T G.991.2] to operate under.
             This object is writable for the CO subtype PMDs (2BaseTL-O).
             It is read-only for the CPE subtype (2BaseTL-R).

          Changing Regional Annex must be performed when the link is
          Down. Attempts to change this object MUST be rejected with
          the error inconsistentValue, if the link is Up or
          Initializing.

             If a Clause 45 MDIO Interface to the PMD is present, then this
             attribute will map to the Region bits in the 2B general
          parameter register"
          REFERENCE
             "[802.3ah] 45.2.1.34"
          ::= { efmCuPmd2BEntry 1 }

      efmCuPmd2BPower  OBJECT-TYPE
          SYNTAX  Integer32(0..15)
          UNITS       "dBm"
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "Desired Signal Transmit Power. Multiple of 0.5dBm.
             This object is writable for the CO subtype PMDs (2BaseTL-O).
             It is read-only for the CPE subtype (2BaseTL-R).

          Changing of the Signal Transmit Power must be performed when the
          link is Down. Attempts to change this object MUST be rejected with
          the error inconsistentValue, if the link is Up or
          Initializing.

             If a Clause 45 MDIO Interface to the PMD is present, then this
             attribute will map to the Power bits in the 2B PMD
```

parameters register"

```
          REFERENCE
             "[802.3ah] 45.2.1.35"
          ::= { efmCuPmd2BEntry 2 }

      efmCuPmd2BDataRate  OBJECT-TYPE
          SYNTAX  Integer32(0..5696)
          UNITS       "Kbps"
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "Desired 2BaseTL PMD Data Rate.
          The rate is fixed when the value is n x 64Kbps, where n=3..60 for
          16-TCPAM and n=12..89 for 32-TCPAM. The value of 0 means
          that data rate is not fixed but is adaptive and should be set to
          the maximum attainable rate during line probing.
             This object is writable for the CO subtype PMDs (2BaseTL-O).
             It is read-only for the CPE subtype (2BaseTL-R).

          Changing of the Data Rate must be performed when the
          link is Down. Attempts to change this object MUST be rejected with
          the error inconsistentValue, if the link is Up or
          Initializing.

             If a Clause 45 MDIO Interface to the PMD is present, then this
             attribute will map to the Data Rate bits in the 2B PMD
          parameters register"
          REFERENCE
             "[802.3ah] 45.2.1.35"
          ::= { efmCuPmd2BEntry 3 }

      efmCuPmd2BConstellation  OBJECT-TYPE
          SYNTAX  INTEGER {
                   tcpam16(1), -- 16-TCPAM
                   tcpam32(2)  -- 32-TCPAM
               }
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
             "Desired TCPAM Constellation of the 2BaseTL PMD.
             This object is writable for the CO subtype PMDs (2BaseTL-O).
             It is read-only for the CPE subtype (2BaseTL-R).

          Changing Constellation must be performed when the link is
          Down. Attempts to change this object MUST be rejected with
          the error inconsistentValue, if the link is Up or
          Initializing.

             If a Clause 45 MDIO Interface to the PMD is present, then this
```

```
            attribute will map to the Constellation bits in the 2B general
        parameter register"
        REFERENCE
            "[802.3ah] 45.2.1.34"
        ::= { efmCuPmd2BEntry 4 }

    efmCuPmd2BLoopAtnThreshold  OBJECT-TYPE
        SYNTAX  Integer32(-127..128)
        UNITS       "dB"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Desired Loop Attenuation Threshold for the 2BaseTL PMD.
             This object configures the loop attenuation alarm threshold.
             When the current value of Loop Attenuation reaches
         or exceeds this threshold, a efmCuPmd2BLoopAttnDefect
         notification MAY be generated.

            This object is writable for the CO subtype PMDs (2BaseTL-O).
            It is read-only for the CPE subtype (2BaseTL-R).

        Changing of the Loop Attenuation Threshold must be performed when the
        link is Down. Attempts to change this object MUST be rejected with
        the error inconsistentValue, if the link is Up or
        Initializing.

            If a Clause 45 MDIO Interface to the PMD is present, then this
            attribute will map to the Loop attenuation threshold bits in the
        2B PMD line quality thresholds register"
        REFERENCE
            "[802.3ah] 45.2.1.36"
        ::= { efmCuPmd2BEntry 5 }

    efmCuPmd2BSnrMgnThreshold  OBJECT-TYPE
        SYNTAX  Integer32(-127..128)
        UNITS       "dB"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Desired SNR Margin Threshold for the 2BaseTL PMD.
             This object configures the SNR margin alarm threshold.
             When the current value of SNR Margin reaches
         or exceeds this threshold, a efmCuPmd2BSnrMgnDefect
         notification MAY be generated.

            This object is writable for the CO subtype PMDs (2BaseTL-O).
            It is read-only for the CPE subtype (2BaseTL-R).
```

```
        Changing of the SNR Margin Threshold must be performed when the
        link is Down. Attempts to change this object MUST be rejected with
        the error inconsistentValue, if the link is Up or
        Initializing.

           If a Clause 45 MDIO Interface to the PMD is present, then this
           attribute will map to the SNR margin threshold bits in the
        2B PMD line quality thresholds register"
        REFERENCE
           "[802.3ah] 45.2.1.36"
        ::= { efmCuPmd2BEntry 6 }



    -- 10PassTS specific PMD group

    efmCuPmd10PTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF EfmCuPmd10PEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
           "Table for EFMCu 10PassTS PMDs (modems)."
         ::= { efmCuPmd 3 }

    efmCuPmd10PEntry OBJECT-TYPE
        SYNTAX  EfmCuPmd10PEntry
        MAX-ACCESS  not-accessible
        STATUS  current
        DESCRIPTION
           "An entry in the EFMCu 10PassTS PMD table."
        AUGMENTS { efmCuPmdEntry }
         ::= { efmCuPmd10PTable 1 }

    EfmCuPmd10PEntry ::=
        SEQUENCE {
            efmCuPmd10PElectricalLength      Integer32
         -- To be continued
            }

    efmCuPmd10PElectricalLength  OBJECT-TYPE
        SYNTAX  Integer32(0..1024)
        UNITS       "m"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "Electrical Length in meters as perceived by the 10PassTS PMD
        after the link is established.
           The value of 0 is returned if the link is Down or Initializing
```

```
          or the PMD is unable to estimate the Electrical Length.

             If a Clause 45 MDIO Interface to the PMD is present, then this
             attribute will map to the 10P Electrical Length register"
          REFERENCE
             "[802.3ah] 45.2.1.21"
          ::= { efmCuPmd10PEntry 1 }


      -- efmCuAvailableStackTable for use in Discovery

      efmCuAvailableStackTable  OBJECT-TYPE
          SYNTAX        SEQUENCE OF EfmCuAvailableStackEntry
          MAX-ACCESS    not-accessible
          STATUS        current
          DESCRIPTION
             "This table, modeled after ifStackTable from [IF-MIB],
          contains information on the possible 'on-top-of'
          relationships between the multiple sub-layers of network
          interfaces (as opposed to actual relationships in
          ifStackTable).  In particular, it contains information on
          which PCS ports (sub-layers) can possible run 'on top of'
          which PMDs (sublayers), as determined by cross-connect
             capability of the EFMCu device, where each sub-layer
             corresponds to a conceptual row in the ifTable.  For example,
          when the PCS port with ifIndex value x can be connected
             to run on top of the PMD with ifIndex value y, then this table
             contains:

               efmCuAvailableStackStatus.x.y=active

             For each ifIndex value, I, which identifies a PCS or PMD
             interface, there are always at least two instantiated rows
             in this table associated with I.  For one of these rows, I
             is the value of efmCuAvailableStackHigherLayer; for the other,
          I is the value of efmCuAvailableStackLowerLayer.
             Note that there's always at least on PCS for each PMD and at
             least one PMD for each PCS in the EFMCu devices.
             This table is ready only as it describes device capability"
          REFERENCE
             "ifStackTable of RFC 2863"
          ::= { efmCuObjects 3 }

      efmCuAvailableStackEntry  OBJECT-TYPE
          SYNTAX        EfmCuAvailableStackEntry
          MAX-ACCESS    not-accessible
          STATUS        current
          DESCRIPTION
```

```
            "Information on a particular relationship between two sub-
            layers, specifying that one sub-layer runs on 'top' of the
            other sub-layer.  Each sub-layer corresponds to a conceptual
            row in the ifTable."
        INDEX {
            efmCuAvailableStackHigherLayer,
            efmCuAvailableStackLowerLayer
            }
        ::= { efmCuAvailableStackTable 1 }

    EfmCuAvailableStackEntry ::=
          SEQUENCE {
              efmCuAvailableStackHigherLayer  InterfaceIndexOrZero,
              efmCuAvailableStackLowerLayer   InterfaceIndexOrZero,
              efmCuAvailableStackStatus       RowStatus
          }

    efmCuAvailableStackHigherLayer  OBJECT-TYPE
        SYNTAX        InterfaceIndexOrZero
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
            "The value of ifIndex corresponding to the higher sub-layer
            of the relationship, i.e., the sub-layer which runs on 'top'
            of the sub-layer identified by the corresponding instance of
            ifStackLowerLayer.  If there is no higher sub-layer (below
            the internetwork layer), then this object has the value 0."
        ::= { efmCuAvailableStackEntry 1 }


    efmCuAvailableStackLowerLayer  OBJECT-TYPE
        SYNTAX        InterfaceIndexOrZero
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
            "The value of ifIndex corresponding to the lower sub-layer
            of the relationship, i.e., the sub-layer which runs 'below'
            the sub-layer identified by the corresponding instance of
            ifStackHigherLayer.  If there is no lower sub-layer, then
            this object has the value 0."
        ::= { efmCuAvailableStackEntry 2 }

    efmCuAvailableStackStatus  OBJECT-TYPE
        SYNTAX         RowStatus
        MAX-ACCESS     read-only
        STATUS         current
        DESCRIPTION
            "The status of the relationship between two sub-layers.
```

```
            This object is read only, unlike ifStackStatus, as it
        describes the device capability."
        ::= { efmCuAvailableStackEntry 3 }


    --
    --     Conformance Statements
    --

    efmCuGroups      OBJECT IDENTIFIER ::= { efmCuConformance 1 }

    efmCuCompliances OBJECT IDENTIFIER ::= { efmCuConformance 2 }

    --     Object Groups

    efmCuPortGroupBasic OBJECT-GROUP
        OBJECTS {
            efmCuPortSidesSupported,
            efmCuPortSide,
            efmCuPAFSupported
            }
        STATUS  current
        DESCRIPTION
           "A collection of objects required for all EFMCu ports."
         ::= { efmCuGroups 1 }

    efmCuGroupPAF OBJECT-GROUP
        OBJECTS {
            efmCuPAFAdminState,
         efmCuPAFDiscoveryCode,
         efmCuPAFRemoteDiscoveryCode,
         efmCuAvailableStackTable
            }
        STATUS  current
        DESCRIPTION
           "A collection of objects that support
           optional Aggregation features on EFMCu ports."
         ::= { efmCuGroups 2 }

    efmCuPmdGroupCommon OBJECT-GROUP
        OBJECTS {
            efmCuPmdRxSnrMgn,
            efmCuPmdRemoteRxSnrMgn
            }
        STATUS  current
        DESCRIPTION
           "A collection of objects that provide
           required information about a 2BaseTL/10PassTS PMD."
```

```
          ::= { efmCuGroups 3 }

      efmCu2BGroup OBJECT-GROUP
          OBJECTS {
              efmCuPmd2BRegion,
              efmCuPmd2BPower,
              efmCuPmd2BDataRate,
              efmCuPmd2BConstellation,
              efmCuPmd2BLoopAtnThreshold,
              efmCuPmd2BSnrMgnThreshold
              }
          STATUS  current
          DESCRIPTION
             "A collection of objects that provide
             required information about a 2BaseTL PMD."
           ::= { efmCuGroups 4 }

      efmCu10PGroup OBJECT-GROUP
          OBJECTS {
              efmCuPmd10PElectricalLength
              }
          STATUS  current
          DESCRIPTION
             "A collection of objects that provide required
             information about a 10PassTS PMD."
           ::= { efmCuGroups 5 }

      -- Compliance Statements

      efmCuCompliance MODULE-COMPLIANCE
          STATUS  current
          DESCRIPTION
             "The compliance statement for 2BaseTL/10PassTS interfaces.
             Compliance with the following external compliance statements
          is prerequisite:

             MIB Module             Compliance Statement
             ----------             --------------------
             IF-MIB                 ifCompliance3
             IF-INVERTED-STACK-MIB  ifInvCompliance
             EtherLike-MIB          dot3Compliance2
             MAU-MIB                mauModIfCompl3"

          MODULE  -- this module
              MANDATORY-GROUPS {
                  efmCuPortGroupBasic,
                  efmCuPmdGroupCommon,
                  efmCuPmd2BGroup,
```

```
                efmCuPmd10PGroup
                }

            OBJECT       efmCuPortSidesSupported
            SYNTAX  INTEGER {
                 subscriber(1),
                 office(2),
             }
            DESCRIPTION
                "Support for values other than subscriber(1),
                and office(2) is not required."

            OBJECT       efmCuPortSide
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required (needed only for ports
             supporting both subscriber and office sides)"

            -- Editor's Note: To be Continued

        ::= { efmCuCompliances 1 }

    END
```

5. Security Considerations

There are number of managed objects defined in this MIB module that have a MAX-ACCESS clause of read-write. Most objects are writeable only when the link is Down. Writing to these objects can have the following potentially disruptive effects on network operation:
o TBD

The user of this MIB module must therefore be aware that support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations.

The readable objects in this MIB module (i.e., those with MAX-ACCESS other than not-accessible) may be considered sensitive in some environments since, collectively, they provide information about the performance of network interfaces and can reveal some aspects of their configuration. In such environments it is important to control even GET and NOTIFY access to these objects and possibly even to encrypt their values when sending them over the network via SNMP.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed

to access and GET/SET (read/change/create/delete) the objects in this MIB module.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

6. Acknowledgments

Not yet.

Normative References

[802.3ah] IEEE, "IEEE Draft P802.3ah/D2.1: 'Draft amendment to - Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications - Media Access Control Parameters, Physical Layers and Management Parameters for subscriber access networks', 07 October 2003.", October 2003.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D. and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

Informative References

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.

[RFC2864] McCloghrie, K. and G. Hanson, "The Inverted Stack Table Extension to the Interfaces Group MIB", RFC 2864, June 2000.

[RFC3636] Flick, J., "Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)", RFC 3636, September 2003.

Author's Address

Edward Beili
Actelis Networks
Bazel 25
Petach-Tikva
Israel

Phone: +972-3-924-3491
EMail: edward.beili@actelis.com

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.