SCTE · ISBE

STANDARDS

Network Operations Subcommittee

AMERICAN NATIONAL STANDARD

ANSI/SCTE 154-3 2018

Encoder MIB

NOTICE

The Society of Cable Telecommunications Engineers (SCTE) / International Society of Broadband Experts (ISBE) Standards and Operational Practices (hereafter called "documents") are intended to serve the public interest by providing specifications, test methods and procedures that promote uniformity of product, interchangeability, best practices and ultimately the long-term reliability of broadband communications facilities. These documents shall not in any way preclude any member or non-member of SCTE•ISBE from manufacturing or selling products not conforming to such documents, nor shall the existence of such standards preclude their voluntary use by those other than SCTE•ISBE members.

SCTE•ISBE assumes no obligations or liability whatsoever to any party who may adopt the documents. Such adopting party assumes all risks associated with adoption of these documents, and accepts full responsibility for any damage and/or claims arising from the adoption of such documents.

Attention is called to the possibility that implementation of this document may require the use of subject matter covered by patent rights. By publication of this document, no position is taken with respect to the existence or validity of any patent rights in connection therewith. SCTE•ISBE shall not be responsible for identifying patents for which a license may be required or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Patent holders who believe that they hold patents which are essential to the implementation of this document have been requested to provide information about those patents and any related licensing terms and conditions. Any such declarations made before or after publication of this document are available on the SCTE•ISBE web site at http://www.scte.org.

CONTENTS

SCOPE
This document is identical to SCTE 154-3 2008 except for informative components which may have been updated such as the title page, NOTICE text, headers and footers. No normative changes have been made to this document.

This document provides the branch object identifiers for each of the MIBs within the SCTE HMS HEADENDIDENT Tree.

COPYRIGHT
The MIB definition found in this document may be incorporated directly in products without further permission from the copyright owner, SCTE.

NORMATIVE REFERENCE
IETF RFC 2578 SNMPv2-SMI
IETF RFC 2579 SNMPv2-TC
IETF RFC 2863 IF-MIB
IETF RFC 4001 INET-ADDRESS-MIB
IETF RFC 4133 ENTITY-MIB
SCTE 36 2002R2007 (formerly HMS028) SCTE-ROOT
SCTE 37 2007 (formerly HMS072) SCTE-HMS-ROOTS

INFORMATIVE REFERENCE
HE MPEG MIBs Overview Document – Number TBD

TERMS AND DEFINITIONS
This document defines the following terms:

Management Information Base (MIB) – the specification of information in a manner that allows standard access through a network management protocol.

REQUIREMENTS
This section defines the mandatory syntax of the SCTE-HMS-HEADENDIDENT-MIB. It follows the IETF Simple Network Management Protocol (SNMP) for defining managed objects.

The syntax is given below.

```
SCTE-HMS-MPEG-ENCODER-MIB DEFINITIONS ::= BEGIN

IMPORTS
      OBJECT-TYPE, MODULE-IDENTITY, TimeTicks,
      Unsigned32, Integer32, enterprises, Counter32
            FROM SNMPv2-SMI
      OBJECT-GROUP, MODULE-COMPLIANCE
            FROM SNMPv2-CONF
      DisplayString
            FROM SNMPv2-TC
      InetAddressPrefixLength, InetPortNumber, InetAddress,
      InetAddressType
            FROM INET-ADDRESS-MIB
      InterfaceIndex
            FROM IF-MIB
      entPhysicalIndex, PhysicalIndex
            FROM ENTITY-MIB
      DeviceEnableDisableValues, MpegErrorStatus,
      HePIDValue,VideoInputFrameRateType
            FROM SCTE-HMS-HEADENDIDENT-TC;

mpegEncoderMIB MODULE-IDENTITY
      LAST-UPDATED "200710031700Z"
      ORGANIZATION
            "SCTE HMS Working Group"
      CONTACT-INFO
            "SCTE HMS Subcommittee, Chairman
             mail to: standards@scte.org"
      DESCRIPTION
            "The MIB module is for representing an MPEG2 or MPEG4-AVC video
            encoder present in the headend (or indoor) and supported by an SNMP
            agent. Multiple encoders sharing a single agent are also supported,
         the encoders being identified by entPhysicalIndex and unique
         ifIndexes for their interfaces.

            This MIB assumes there is one video source per encoder.
            If a physical unit encodes more than one video source at
            a time it shall be considered to be a group of encoders sharing
            a common chassis and each encoder treated as a discrete  physical
            entity.

            An encoder may encode more than one stream from a single source."

      REVISION "200710031700Z"
      DESCRIPTION
            " Added SCTE-HMS-HEADENDIDENT-TC name to file"

      REVISION "200709281545Z"
      DESCRIPTION
            " Prepare MIB for ballot"

            ::= { enterprises scteRoot(5591) scteHmsTree (1) insidePlantIdent (11)
            heDigital (5) heDigitalMpegGroups (5) 1}


mpegEncoderMIBObjects  OBJECT IDENTIFIER ::= { mpegEncoderMIB 1 }
configurationReport  OBJECT IDENTIFIER ::= { mpegEncoderMIBObjects 1 }
```

```
inputMonitor          OBJECT IDENTIFIER ::= { mpegEncoderMIBObjects 2 }
outputMonitor         OBJECT IDENTIFIER ::= { mpegEncoderMIBObjects 3 }

-- Conformance Information

mpegEncoderMIBConformance  OBJECT IDENTIFIER ::= { mpegEncoderMIB 2 }

-- The minimum compliance statement for MPEG encoders.

mpegEncoderMIBCompliances  OBJECT IDENTIFIER ::= { mpegEncoderMIBConformance 1 }
mpegEncoderMIBGroups  OBJECT IDENTIFIER ::= { mpegEncoderMIBConformance 2 }


-- *********************************************************************
-- ******************     CONFIGURATION REPORT     *********************
-- *********************************************************************

-- The tables in this section define the output from the encoder. Each
-- component is referenced to the service to which it belongs. Each
-- service is referenced to the transport stream in which it is carried
-- and each transport stream is referenced to the physical and logical
-- output.
-- If a component is used in more than one service then the table row
-- should be repeated with the service cross reference pointing to the
-- alternative service. Likewise services output on more than one
-- transport streams should have a table row for each instance with
-- appropriate referencing to each transport stream.

encoderCfgVideoTable  OBJECT-TYPE
     SYNTAX SEQUENCE OF EncoderCfgVideoEntry
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
          "This table shows the video configuration of this encoder."
     ::= { configurationReport 1 }

encoderCfgVideoEntry  OBJECT-TYPE
     SYNTAX      EncoderCfgVideoEntry
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
          "An entry containing management information applicable to a
          particular encoder's video configuration.  This table is indexed by
          the entPhysicalIndex of the encoder, and encoderCfgVideoIndex
          to allow for multiple video channels. All video channels shall be
        derived from a single source and consist of a primary and auxiliary
        videos such as PIP. If more than one primary video is encoded within
        the chassis then it shall be treated as a multiple encoder chassis."
     INDEX { entPhysicalIndex,
       encoderCfgVideoIndex }
     ::= { encoderCfgVideoTable 1 }

EncoderCfgVideoEntry ::= SEQUENCE {
   encoderCfgVideoIndex
          Unsigned32,
   encoderCfgVideoType
          INTEGER,
```

```
        encoderCfgVideoCompression
                INTEGER,
        encoderCfgVideoPid
                HePIDValue,
        encoderCfgVideoPcrPid
                HePIDValue,
        encoderCfgVideoServiceIndex
                Unsigned32,
        encoderCfgVideoVertResolution
                Integer32,
        encoderCfgVideoHorzResolution
                Integer32,
        encoderCfgVideoBitrateAvg
                Integer32,
        encoderCfgVideoBitrateMax
                Integer32,
        encoderCfgVideoBitrateMin
                Integer32,
        encoderCfgVideoFilmMode
                DeviceEnableDisableValues ,
        encoderCfgVideoRateMode
                INTEGER,
        encoderCfgVideoBorderProcessing
                INTEGER,
        encoderCfgVideoPesAlignment
                INTEGER,
        encoderCfgVideoCodingDelay
                Integer32,
        encoderCfgVideoDeblockEnable
                DeviceEnableDisableValues,
        encoderCfgVideoDeblockAlpha
                Integer32,
        encoderCfgVideoDeblockBeta
                Integer32,
        encoderCfgVideoIdrRate
                Integer32,
        encoderCfgVideoInputIf
                INTEGER,
        encoderCfgVideoInputFrameRate
                VideoInputFrameRateType,
        encoderCfgVideoInputScan
                INTEGER,
        encoderCfgVideoInputFormat
                INTEGER,
        encoderCfgVideoAspectRatio
                INTEGER
}

encoderCfgVideoIndex  OBJECT-TYPE
        SYNTAX     Unsigned32
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                "Identifier index for each video channel (from a single source)
            provided by this encoder."
        ::= { encoderCfgVideoEntry 1 }
```

```
encoderCfgVideoType  OBJECT-TYPE
     SYNTAX     INTEGER {
          other (1),
          primary (2),
          pip (3),
          dvbH (4)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "If the video is encoded into more than one component this object
          defines the type of this component. If one video component
          encoded then this object shall always be primary.
               mpeg2 - ISO/IEC 13818 part 2 video
               mpeg4 - ISO/IEC 14496 part 2 video
               avc   - ITU-T H.264 video"
     ::= { encoderCfgVideoEntry 2 }


encoderCfgVideoCompression  OBJECT-TYPE
     SYNTAX     INTEGER {
          other (1),
          mpeg2 (2),
          mpeg4 (3),
          avc (4)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Video encoding standard
               mpeg2 - ISO/IEC 13818 part 2 video
               mpeg4 - ISO/IEC 14496 part 2 video
               avc   - ITU-T H.264 video"
     ::= { encoderCfgVideoEntry 3 }


encoderCfgVideoPid  OBJECT-TYPE
     SYNTAX     HePIDValue
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "The PID carrying the primary video component."
     ::= { encoderCfgVideoEntry 4 }


encoderCfgVideoPcrPid  OBJECT-TYPE
     SYNTAX     HePIDValue
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "The PID of the PCR used by this component."
     ::= { encoderCfgVideoEntry 5 }


encoderCfgVideoServiceIndex  OBJECT-TYPE
     SYNTAX     Unsigned32
```

```
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "When multiple services/transport streams are supported
            by this encoder, this field provides a cross reference to
            a specific service based on the encoderCfgServiceInstIndex
            in the encoderCfgServiceTable.
            When this component is output as an elementary stream
            without PAT/PMT reference, this field provides an indirect
            cross reference to the output transport stream"
        ::= { encoderCfgVideoEntry 6 }


encoderCfgVideoVertResolution  OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Number of active video lines used in encoding.
            Values include, but are not restricted to
            240, 288, 480, 512, 576, 608, 720 and 1080"
        ::= { encoderCfgVideoEntry 7 }


encoderCfgVideoHorzResolution  OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Number of pixels per line used in encoding.
            Values include, but are not restricted to
          352, 480, 528, 544, 640, 704, 720, 1280, 1440, 1920"
        ::= { encoderCfgVideoEntry 8 }


encoderCfgVideoBitrateAvg  OBJECT-TYPE
        SYNTAX      Integer32
        UNITS        "bps"
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Average or nominal bit rate for video component expressed in
          Transport rate. If component is output as elementary stream
          The encoder shall convert the actual rate to equivalent
          hypothetical transport rate for compatibility with down stream
          monitoring points "
        ::= { encoderCfgVideoEntry 9 }


encoderCfgVideoBitrateMax  OBJECT-TYPE
        SYNTAX      Integer32
        UNITS          "bps"
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Maximum allowed bit rate for video component when
            stat mux/VBR selected. Equal to average when CBR."
```

```
        ::= { encoderCfgVideoEntry 10 }


encoderCfgVideoBitrateMin  OBJECT-TYPE
     SYNTAX     Integer32
     UNITS              "bps"
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Minimum allowed bit rate for video component when
          stat mux/VBR selected. Equal to average when CBR."
     ::= { encoderCfgVideoEntry 11 }


encoderCfgVideoFilmMode  OBJECT-TYPE
     SYNTAX     DeviceEnableDisableValues
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Normal frame encoding or 3-2 pulldown allowed."
     ::= { encoderCfgVideoEntry 12 }


encoderCfgVideoRateMode  OBJECT-TYPE
     SYNTAX     INTEGER {
          cbr (1),
          vbr (2)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Fixed bit rate, stat mux or other variable rate mode selected."
     ::= { encoderCfgVideoEntry 14 }


encoderCfgVideoBorderProcessing  OBJECT-TYPE
     SYNTAX     INTEGER {
          disabled (1),
          cropped (2),
          other(3)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Border processing selected."
     ::= { encoderCfgVideoEntry 15 }


encoderCfgVideoPesAlignment  OBJECT-TYPE
     SYNTAX     INTEGER {
          other     (1),
          aligned   (2),
          nonAligned (3)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
```

```
                "PES to transport stream alignment."
        ::= { encoderCfgVideoEntry 16 }


encoderCfgVideoCodingDelay  OBJECT-TYPE
        SYNTAX      Integer32
        UNITS           "milliseconds"
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Video delay due to encoding, frame capture to PTS.
          Zero indicates parameter not reported."
        ::= { encoderCfgVideoEntry 17 }


encoderCfgVideoDeblockEnable  OBJECT-TYPE
        SYNTAX      DeviceEnableDisableValues
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Deblock filter enable/disable."
        ::= { encoderCfgVideoEntry 18 }


encoderCfgVideoDeblockAlpha  OBJECT-TYPE
        SYNTAX      Integer32 (-6..6)
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Deblock filter strength - edge offset."
        ::= { encoderCfgVideoEntry 19 }


encoderCfgVideoDeblockBeta  OBJECT-TYPE
        SYNTAX      Integer32 (-6..6)
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Deblock filter strength - edge neighbor offset."
        ::= { encoderCfgVideoEntry 20 }


encoderCfgVideoIdrRate  OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Rate IDR pictures are sent, expressed in frames.Say AVC"
        ::= { encoderCfgVideoEntry 21 }


encoderCfgVideoInputIf  OBJECT-TYPE
        SYNTAX      INTEGER {
            other (1),
            analog (2),
            sdi (3),
            hdsdi (4)    }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Video input standard selected."
        ::= { encoderCfgVideoEntry 22 }
```

```
encoderCfgVideoInputFrameRate  OBJECT-TYPE
     SYNTAX      VideoInputFrameRateType
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
           "Video input rate selected. autoSelect indicates encoder
         can accept most frame rates without specific selection.
         f29or30Hz and f59or60Hz indicate limited auto-select
         between 29.97Hz and 30Hz or 59.94Hz and 60Hz supported
         by the encoder."
     ::= { encoderCfgVideoEntry 23 }

encoderCfgVideoInputScan  OBJECT-TYPE
     SYNTAX      INTEGER {
           interlaced (1),
           progressive (2)    }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
           "Video input scanning selected."
     ::= { encoderCfgVideoEntry 24 }

encoderCfgVideoInputFormat  OBJECT-TYPE
     SYNTAX      INTEGER {
           component (1),
           composite (2)      }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
           "Video input format selected."
     ::= { encoderCfgVideoEntry 25 }

encoderCfgVideoAspectRatio  OBJECT-TYPE
     SYNTAX      INTEGER {
           other (1),
           aspect4x3 (2),
           aspect16x9 (3)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
           "Input video aspect ratio"
     ::= { encoderCfgVideoEntry 26 }


encoderCfgAudioTable  OBJECT-TYPE
     SYNTAX SEQUENCE OF EncoderCfgAudioEntry
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
           "This table shows the audio configuration of this encoder."
     ::= { configurationReport 2 }

encoderCfgAudioEntry  OBJECT-TYPE
     SYNTAX      EncoderCfgAudioEntry
     MAX-ACCESS not-accessible
     STATUS      current
```

```
        DESCRIPTION
            "An entry containing management information applicable to a
            particular encoder's audio components.  This table is indexed
          by the entPhysicalIndex of the encoder and the
            encoderCfgAudioChannelIndex for each audio component carried by
            this encoder."
        INDEX { entPhysicalIndex,
            encoderCfgAudioChannelIndex }
        ::= { encoderCfgAudioTable 1 }

EncoderCfgAudioEntry ::= SEQUENCE {
        encoderCfgAudioChannelIndex
            Unsigned32,
        encoderCfgAudioEnabled
            DeviceEnableDisableValues,
        encoderCfgAudioStandard
            INTEGER,
        encoderCfgAudioPid
            HePIDValue,
        encoderCfgAudioPcrPid
            HePIDValue,
        encoderCfgAudioServiceIndex
           Unsigned32,
        encoderCfgAudioMode
            INTEGER,
     encoderCfgAudioPassThru
            INTEGER,
        encoderCfgAudioRate
            Integer32,
        encoderCfgAudioLanguageA
            DisplayString,
        encoderCfgAudioLanguageB
            DisplayString,
        encoderCfgAudioLipSync
            INTEGER,
        encoderCfgAudioDialogNorm
            Integer32,
        encoderCfgAudioInputType
            INTEGER,
        encoderCfgAudioInputLabel
            DisplayString

}

encoderCfgAudioChannelIndex  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
            "Identifier index for each audio output channel provided by
           this encoder. This number shall be unique within an encoder."
        ::= { encoderCfgAudioEntry 1 }

encoderCfgAudioEnabled  OBJECT-TYPE
        SYNTAX      DeviceEnableDisableValues
        MAX-ACCESS read-only
        STATUS      current
```

```
            DESCRIPTION
                "Shows if this audio channel is configured for service."
            ::= { encoderCfgAudioEntry 2 }


encoderCfgAudioStandard  OBJECT-TYPE
        SYNTAX     INTEGER {
                other (1),
                ac3 (2),
                eac3 (3),
                mpeg1 (4),
                mpeg2 (5),
                aacMpeg2 (6),
                heAacMpeg2 (7),
                aacMpeg4 (8),
                heAacMpeg4 (9)
        }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "Audio encoding standard for this channel. Values identified
                against 13818-1 stream_types as follows:
          ac3           stream_type 0x81/0x06(DVB)
          eac3          stream_type 0x87
          mpeg1         stream_type 0x03
             mpeg2            stream_type 0x04
          aacMpeg2  stream_type 0x0F
          heAacMpeg2      stream_type 0x0F
          aacMpeg4  stream_type 0x11
             heAacMpeg4  stream_type 0x11"
        ::= { encoderCfgAudioEntry 3 }


encoderCfgAudioPid  OBJECT-TYPE
        SYNTAX     HePIDValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "Selected PID for this audio component."
        ::= { encoderCfgAudioEntry 4 }


encoderCfgAudioPcrPid  OBJECT-TYPE
        SYNTAX     HePIDValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "The PCR used by this component."
        ::= { encoderCfgAudioEntry 5 }


encoderCfgAudioServiceIndex  OBJECT-TYPE
        SYNTAX     Unsigned32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "When multiple services/transport streams are supported
                by this encoder, this field provides a cross reference to
                a specific service based on the encoderCfgServiceInstIndex
                in the encoderCfgServiceTable.
                When this component is output as an elementary stream
```

```
                without PAT/PMT reference, this field provides an indirect
                cross reference to the output transport stream"
        ::= { encoderCfgAudioEntry 6 }


encoderCfgAudioMode  OBJECT-TYPE
        SYNTAX     INTEGER {
                other (1),
                stereo (2),
                joint (3),
                mono (4),
                monoL (5),
                monoR (6),
                dualMono (7),
                fivePlusOne (8),
                unknown (9)
        }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "Selected audio mode for this channel."
        ::= { encoderCfgAudioEntry 7 }

encoderCfgAudioPassThru  OBJECT-TYPE
        SYNTAX     INTEGER {
                encoded (1),
                passThrough (2)    }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "Selected audio mode for this channel."
        ::= { encoderCfgAudioEntry 8 }

encoderCfgAudioRate  OBJECT-TYPE
        SYNTAX     Integer32
        UNITS      "kbps"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "Audio component bit rate expressed in transport rate."
        ::= { encoderCfgAudioEntry 9 }

encoderCfgAudioLanguageA  OBJECT-TYPE
        SYNTAX     DisplayString
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "ISO 693 language code for this audio channel, or for 1st mono when
                this channel carries a dual mono pair"
        REFERENCE
                "ISO 639-2:1998 Codes for the representation of names of languages
                -- Part 2: Alpha-3 code International Organization for
                Standardization"
        ::= { encoderCfgAudioEntry 10 }

encoderCfgAudioLanguageB  OBJECT-TYPE
        SYNTAX     DisplayString
```

```
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "ISO 693 language code for 2nd mono when this channel carries a dual
            mono pair. Returns an empty string when not applicable."
        REFERENCE
            "ISO 639-2:1998 Codes for the representation of names of languages
            -- Part 2: Alpha-3 code International Organization for
            Standardization"
        ::= { encoderCfgAudioEntry 11 }

encoderCfgAudioLipSync  OBJECT-TYPE
        SYNTAX      INTEGER {
            fixed (1),
            manual (2)
        }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Manual override of default lip sync delay"
        ::= { encoderCfgAudioEntry 12 }


encoderCfgAudioDialogNorm  OBJECT-TYPE
        SYNTAX      Integer32 (-31..0)
        UNITS       "dBFS"
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            " This object is only relevant to AC-3, its value
          will be fixed at 0 (not indicated) for all other standards.
          Level for AC-3 dialogue normalization. When this object
          is applicable range is -31 to -1. "
        ::= { encoderCfgAudioEntry 13 }

encoderCfgAudioInputType  OBJECT-TYPE
        SYNTAX      INTEGER {
            embedded (1),
            aes (2),
            analog (3)
        }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Format of input audio for this channel"
        ::= { encoderCfgAudioEntry 14 }

encoderCfgAudioInputLabel  OBJECT-TYPE
        SYNTAX      DisplayString
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Input interface number for this channel, if a physical interface
            the designation printed on the connector label, if an embedded pair
            1/2 for the first pair, 3/4 for the second pair etc."
        ::= { encoderCfgAudioEntry 15 }
```

```
 encoderCfgVbiTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF EncoderCfgVbiEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
           "This table shows the configuration of this encoder for processing
           data from the vertical blanking interval (VBI)."
      ::= { configurationReport 3 }

encoderCfgVbiEntry  OBJECT-TYPE
      SYNTAX     EncoderCfgVbiEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
           "An entry containing management information applicable to a
           particular encoder's vertical blanking interval (VBI) specific
           configuration."
      INDEX { entPhysicalIndex,
          encoderCfgVbiIndex }
      ::= { encoderCfgVbiTable 1 }

EncoderCfgVbiEntry ::= SEQUENCE {
      encoderCfgVbiIndex
           Unsigned32,
      encoderCfgVbiField
           INTEGER,
      encoderCfgVbiLine
           Integer32,
      encoderCfgVbiType
           INTEGER,
      encoderCfgVbiCarriage
           INTEGER,
      encoderCfgVbiCompIndex
           Unsigned32
}

encoderCfgVbiIndex  OBJECT-TYPE
      SYNTAX     Unsigned32
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
           "Serial index for each row of this table. Each row represents
          one line of VBI. No dependency between line index and actual
          line number is implied."
      ::= { encoderCfgVbiEntry 1 }

encoderCfgVbiField  OBJECT-TYPE
      SYNTAX     INTEGER {
           field1 (1),
           field2 (2),
           both (3)    }
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "The source field configured for this line of VBI."
```

```
      ::= { encoderCfgVbiEntry 2 }


encoderCfgVbiLine  OBJECT-TYPE
      SYNTAX      Integer32
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
            "The source line number configured for this line of VBI.
            Range 10..22 for NTSC f1/f2 convention, 6-23 and 318-335
         for PAL convention. "
      ::= { encoderCfgVbiEntry 3 }


encoderCfgVbiType  OBJECT-TYPE
      SYNTAX      INTEGER {
            other (1),    eia608Caption(2),
            amol (3),     amol2(4),
            nabts (5),    vitc(6),
            vits (7),     tvGuide(8),
            cgmsA (9),    dataBcast(10),
            wst (11),     vps (12),
            wss (13),     epg (14),
            barData(15),  scte104(16)}
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
            "The type of VBI service carried on this line of VBI."
      ::= { encoderCfgVbiEntry 4 }


encoderCfgVbiCarriage  OBJECT-TYPE
      SYNTAX      INTEGER {
            otherInVideo (1), otherInPid (2),
            scte20 (3),           scte21 (4),
            scte127 (5),          etsiEn300472 (6),
            etsiEn301775 (7)   }
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
            "The means of carriage configured for this line of VBI.
         Scte20, scte21 and otherInVideo are all VBI carried in
         video, otherInPid, scte127, etsiEn300472 and
         etsiEn301775 are all VBI carried in its own PID"
      ::= { encoderCfgVbiEntry 5 }



encoderCfgVbiCompIndex  OBJECT-TYPE
      SYNTAX      Unsigned32
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
            "If this VBI is carried in its own PID this field provides
            a cross reference to its definition in the encoderCfgAncilTable.
            If this VBI is carried in the video this field provides
            a cross reference to the video component in the
            encoderCfgVideoTable."
      ::= { encoderCfgVbiEntry 6 }
```

```
encoderCfgVancTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF EncoderCfgVancEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
            "This table shows the configuration of this encoder for processing
            data from the vertical ancillary space (VANC)."
      ::= { configurationReport 4 }

encoderCfgVancEntry  OBJECT-TYPE
      SYNTAX      EncoderCfgVancEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
            "An entry containing management information applicable to a
            particular encoder's vertical ancillary data (VANC) specific
            configuration."
      INDEX { entPhysicalIndex,
          encoderCfgVancServiceIndex }
      ::= { encoderCfgVancTable 1 }

EncoderCfgVancEntry ::= SEQUENCE {
      encoderCfgVancServiceIndex
            Unsigned32,
      encoderCfgVancDid
            Integer32,
      encoderCfgVancSdid
            Integer32,
      encoderCfgVancType
            INTEGER,
      encoderCfgVancCarriage
            INTEGER,
      encoderCfgVancCompIndex
            Unsigned32
}

encoderCfgVancServiceIndex  OBJECT-TYPE
      SYNTAX     Unsigned32
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
            "Serial index for each row of this table. Each row represents
            one service extracted from VANC."
      ::= { encoderCfgVancEntry 1 }

encoderCfgVancDid  OBJECT-TYPE
      SYNTAX     Integer32 (0..1023)
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
            "Identifier for this VANC service type."
      ::= { encoderCfgVancEntry 2 }


encoderCfgVancSdid  OBJECT-TYPE
      SYNTAX     Integer32 (0..1023)
      MAX-ACCESS read-only
```

```
        STATUS      current
        DESCRIPTION
            "Secondary identifier for this VANC service type."
        ::= { encoderCfgVancEntry 3 }


encoderCfgVancType  OBJECT-TYPE
        SYNTAX      INTEGER {
            other (1),              cc608(2),
            cc708 (3),              atvef(4),
            progDescription (5),    dataBroadcast(6),
            nabts (7),              guidePlus (8),
            amol (9),               amol2 (10)    }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "The type of VANC service carried on this line of VANC."
        ::= { encoderCfgVancEntry 4 }


encoderCfgVancCarriage  OBJECT-TYPE
        SYNTAX      INTEGER {
            inVideo (1),
            inOwnPid (2) }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "The means of carriage configured for the output of
          this VANC component."
        ::= { encoderCfgVancEntry 5 }


encoderCfgVancCompIndex  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
          "If this VBI is carried in its own PID this field provides
          a cross reference to its definition in the encoderCfgAncilTable.
          If this VBI is carried in the video this field provides
          a cross reference to the video component in the
          encoderCfgVideoTable."
        ::= { encoderCfgVancEntry 6 }


encoderCfgAncilTable    OBJECT-TYPE
        SYNTAX          SEQUENCE  OF  EncoderCfgAncilEntry
        MAX-ACCESS      not-accessible
        STATUS          current
        DESCRIPTION
            "This table shows the configuration for all other components
            supported by this encoder."
        ::=  { configurationReport 5 }


encoderCfgAncilEntry  OBJECT-TYPE
        SYNTAX      EncoderCfgAncilEntry
```

```
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
               "An entry containing management information applicable to a
               particular encoder's configuration for specific ancillary
             components.  This table is indexed by the entPhysicalIndex of
             the encoder and the encoderCfgAncilComponentIndex for each
             ancillary component carried by this encoder."
          INDEX { entPhysicalIndex,
               encoderCfgAncilComponentIndex }
          ::= { encoderCfgAncilTable 1 }


EncoderCfgAncilEntry ::= SEQUENCE {
          encoderCfgAncilComponentIndex
               Unsigned32,
          encoderCfgAncilEnabled
               DeviceEnableDisableValues,
          encoderCfgAncilType
               INTEGER,
          encoderCfgAncilSourceType
               INTEGER,
          encoderCfgAncilPid
               HePIDValue,
          encoderCfgAncilPcrPid
               HePIDValue,
          encoderCfgAncilRate
               Integer32,
          encoderCfgAncilLanguage
               DisplayString,
          encoderCfgAncilServiceIndex
               Unsigned32
          }

encoderCfgAncilComponentIndex  OBJECT-TYPE
          SYNTAX     Unsigned32
          MAX-ACCESS not-accessible
          STATUS     current
          DESCRIPTION
               "Identifier index for each ancillary output channel provided by this
               encoder. Corresponds with encoderInputMonAncilCfgChannel in input
               monitor. This number shall be unique within an encoder."
          ::= { encoderCfgAncilEntry 1 }

encoderCfgAncilEnabled  OBJECT-TYPE
          SYNTAX     DeviceEnableDisableValues
          MAX-ACCESS read-only
          STATUS     current
          DESCRIPTION
               "Shows if this ancillary channel is configured for service."
          ::= { encoderCfgAncilEntry 2 }

encoderCfgAncilType  OBJECT-TYPE
          SYNTAX     INTEGER {
               other (1),
               scte35 (2),
               subtitles (3),
               codeDownload (4),
```

```
                dsmcc (5),
                epg (6),
                eia708 (7)
        }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "The type of service provided by this ancillary component.
          When ancillary component is carrying data defined in
          encoderCfgVbiTable or encoderCfgVancTable then it is
          sufficient to set this entry to other."
        ::= { encoderCfgAncilEntry 3 }

encoderCfgAncilSourceType  OBJECT-TYPE
        SYNTAX     INTEGER {
                other (1),
                physical (2),
                ip (3),
                vbi (4),
                vanc (5)
        }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "The type of input used by this ancillary component.
          If VBI or VANC further information provided in specific
          tables above."
        ::= { encoderCfgAncilEntry 4 }


encoderCfgAncilPid  OBJECT-TYPE
        SYNTAX     HePIDValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "Selected PID for this ancillary component.
          If the data defined in this row is combined with
          another element e.g. EIA708 data input via SMPTE-333
          and carried in video, then this and other relevant
          entries need to duplicate the values applicable to
          the carrying component."
        ::= { encoderCfgAncilEntry 5 }


encoderCfgAncilPcrPid  OBJECT-TYPE
        SYNTAX     HePIDValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "The PID of PCR used by this component."
        ::= { encoderCfgAncilEntry 6 }


encoderCfgAncilRate  OBJECT-TYPE
        SYNTAX     Integer32
        UNITS         "bps"
        MAX-ACCESS read-only
```

```
        STATUS      current
        DESCRIPTION
             "The bit rate of this ancillary component. For components
             transmitted sporadically such as SCTE-35 this will be the average
             rate calculated over the period since this value was last read. In
             cases where packets were sent but the period is so long that the
             resulting rate is less 1, the result shall be rounded up to 1 to
             indicate data has been present."
        ::= { encoderCfgAncilEntry 7 }


encoderCfgAncilLanguage  OBJECT-TYPE
        SYNTAX      DisplayString
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
             "If a language is defined for this component, the ISO 693 language
             code for this ancillary component. If not defined this entry shall
             be null."
        REFERENCE
             "ISO 639-2:1998 Codes for the representation of names of languages
             -- Part 2: Alpha-3 code International Organization for
             Standardization"
        ::= { encoderCfgAncilEntry 8 }


encoderCfgAncilServiceIndex  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
             "When multiple services/transport streams are supported
             by this encoder, this field provides a cross reference to
             a specific service based on the encoderCfgServiceInstIndex
             in the encoderCfgServiceTable.
             When this component is output as an elementary stream
             without PAT/PMT reference, this field provides an indirect
             cross reference to the output transport stream"
        ::= { encoderCfgAncilEntry 9 }


encoderCfgServiceTable  OBJECT-TYPE
        SYNTAX SEQUENCE OF EncoderCfgServiceEntry
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
             "This table shows the configuration the service or services output
             by this encoder"
        ::= { configurationReport 6 }

encoderCfgServiceEntry  OBJECT-TYPE
        SYNTAX      EncoderCfgServiceEntry
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
             "An entry containing management information applicable to a
             particular encoder's service configuration.  This table is indexed
```

```
            by the entPhysicalIndex of the encoder and the
            encoderCfgServiceInstIndex for each Service stream output
by this encoder. If the encoder outputs elementary streams only,
then this table shall remain empty."
        INDEX { entPhysicalIndex,
            encoderCfgServiceInstIndex }
        ::= { encoderCfgServiceTable 1 }


EncoderCfgServiceEntry ::= SEQUENCE {
      encoderCfgServiceInstIndex
            Unsigned32,
      encoderCfgServiceType
            INTEGER,
      encoderCfgServiceId
            Integer32,
      encoderCfgServiceName
            DisplayString,
      encoderCfgServiceTransportIndex
            Integer32
}

encoderCfgServiceInstIndex  OBJECT-TYPE
      SYNTAX     Unsigned32
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
            "Identifier index for each service provided by this encoder."
      ::= { encoderCfgServiceEntry 1 }

encoderCfgServiceType  OBJECT-TYPE
      SYNTAX     INTEGER {
            other (1),
            noService (2),
            primary (3),
            audio (4),
            data (5),
            codeDownload (6),
            pip (7),
            epg (8)
      }
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
            "Service type. Note: the primary service is the service carrying
            the video and associated components, an audio service may include
            data and other components.
         noService indicates the elementary streams referencing this table
         row are not contained in a service i.e. they do not have PAT/PMT
         reference. In this case this row is used only to cross reference
         the elementary stream to the transport stream carrying it and does
         not represent a service/MPEG2 program."
      ::= { encoderCfgServiceEntry 2 }

encoderCfgServiceId  OBJECT-TYPE
      SYNTAX     Integer32 (-1..65535)
      MAX-ACCESS read-only
      STATUS     current
```

```
        DESCRIPTION
            "If this row represents an actual service this fields shall
            contain its MPEG2 program number.
          If encoderCfgServiceType = noService this field shall be
          set to -1."
        ::= { encoderCfgServiceEntry 3 }


encoderCfgServiceName  OBJECT-TYPE
        SYNTAX    DisplayString
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A name to identify this service. Should be the same as service name
            used in SI if present."
        ::= { encoderCfgServiceEntry 6 }


encoderCfgServiceTransportIndex  OBJECT-TYPE
        SYNTAX    Integer32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "Cross reference to the transport stream index used in the
            encoderCfgTransportTable to identify the transport stream
            carrying this service."
        ::= { encoderCfgServiceEntry 7 }


encoderCfgTransportTable  OBJECT-TYPE
        SYNTAX SEQUENCE OF EncoderCfgTransportEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "This table shows the configuration for the transport stream or
            streams output by this encoder."
        ::= { configurationReport 7 }


encoderCfgTransportEntry  OBJECT-TYPE
        SYNTAX     EncoderCfgTransportEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "An entry containing management information applicable to a
            particular encoder's transport configuration.  This table is
          indexed by the entPhysicalIndex of the encoder and
            an index used to indicate individual transport streams output
          by this encoder."
        INDEX { entPhysicalIndex,
            encoderCfgTransportChannelIndex }
        ::= { encoderCfgTransportTable 1 }


EncoderCfgTransportEntry ::= SEQUENCE {
        encoderCfgTransportChannelIndex
            Unsigned32,
        encoderCfgTransportIfIndex
            InterfaceIndex,
        encoderCfgTransportEnabled
            DeviceEnableDisableValues,
        encoderCfgTransportType
```

```
                    INTEGER,
        encoderCfgTransportId
                Integer32,
        encoderCfgTransportSiStd
                INTEGER,
        encoderCfgTransportRate
                Integer32,
        encoderCfgTransportRateMode
                INTEGER,
        encoderCfgTransportInetAddrType
                InetAddressType,
        encoderCfgTransportDestIpAddr
                InetAddress,
        encoderCfgTransportDestUdpPort
                InetPortNumber,
        encoderCfgTransportGatewayAddr
                InetAddress,
        encoderCfgTransportSubnetMask
                InetAddressPrefixLength,
        encoderCfgTransportIpFec
                INTEGER,
     encoderCfgTransportIpInterleave
                INTEGER,
        encoderCfgTransportIpAddrgMode
                INTEGER
}


encoderCfgTransportChannelIndex  OBJECT-TYPE
     SYNTAX     Unsigned32
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
          "Identifier index for each transport stream output by this encoder.
          The transport stream may be on its own physical interface or sharing
          a physical interface with other transport streams from this encoder."
     ::= { encoderCfgTransportEntry 1 }

encoderCfgTransportIfIndex  OBJECT-TYPE
     SYNTAX     InterfaceIndex
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Cross reference to the physical interface used by this
        transport stream."
     ::= { encoderCfgTransportEntry 2 }


encoderCfgTransportEnabled  OBJECT-TYPE
     SYNTAX     DeviceEnableDisableValues
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Shows if this transport stream is configured for service.
        Note. This is intended for encoders with more than one
        transport stream output and allows the monitor system
        to identify which outputs are in use."
```

```
      ::= { encoderCfgTransportEntry 3 }



encoderCfgTransportType  OBJECT-TYPE
     SYNTAX     INTEGER {
          other (1),        asi (2),
          dhei (3),         spmte310 (4),
          ds3 (5),          ip (6),
       idp (7),        ipRtp (8),
       ipUdp(9),       modulated (10)
                                     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
         " Output type
       asi      Asychonous Serial Interface EN50083-9
       dhei     Digital Head-end Expansion Interface
       smpte-310 Synchronous Serial Interface
       ds3      ANSI T1.404 and variants
       ip Internet Protocol (non-specific)
       idp      Internet Datagram Protocol?????
       ipRtp    Real Time Protocol over IP
       ipUdp    User Datagram Protocol over IP
       modulated      Data modulated on RF carrier
        ."
     ::= { encoderCfgTransportEntry 4 }



encoderCfgTransportId  OBJECT-TYPE
     SYNTAX     Integer32
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
         "MPEG2 transport_stream_id per 13818-1 for this
       transport stream."
     ::= { encoderCfgTransportEntry 5 }



encoderCfgTransportSiStd  OBJECT-TYPE
     SYNTAX     INTEGER {
          other (1),
          mpeg2 (2),
          atsc (3),
          dvb (4),
          dcii (5),
          atscDvb (6),
          noTables (7)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
         "PSI and SI standard provided on this transport stream."
     ::= { encoderCfgTransportEntry 6 }



encoderCfgTransportRate  OBJECT-TYPE
     SYNTAX     Integer32
```

```
        UNITS                "bps"
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Transport stream bit output rate (information rate).
          If constant bit rate (CBR) this is the rate configured.
          If variable bit rate (VBR) this is the maximum rate allowed
          by the configuration. "
        ::= { encoderCfgTransportEntry 7 }


encoderCfgTransportRateMode  OBJECT-TYPE
        SYNTAX      INTEGER {
            cbr (1),
            vbr (2)
        }
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Constant or variable transport rate."
        ::= { encoderCfgTransportEntry 8 }


encoderCfgTransportInetAddrType  OBJECT-TYPE
        SYNTAX      InetAddressType
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            " Address type used for transport stream addressing based on
            InetAddressType from the INET-ADDRESS-MIB to allow for IP v4 and
            IP v6 addressing. Relevant values for this object are
                            unknown(1),
                            ipv4(2),
                            ipv6(3)"
        ::= { encoderCfgTransportEntry 9 }


encoderCfgTransportDestIpAddr  OBJECT-TYPE
        SYNTAX      InetAddress
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Destination IP address for transport stream if over IP.  If not
            over IP value shall be all zeros. May be IPv4 or IPv6 address
            depending on encoderCfgTransportInetAddrType."
        ::= { encoderCfgTransportEntry 10 }


encoderCfgTransportDestUdpPort  OBJECT-TYPE
        SYNTAX      InetPortNumber
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
            "Destination UDP port for transport stream if over IP."
        ::= { encoderCfgTransportEntry 11 }
```

```
encoderCfgTransportGatewayAddr   OBJECT-TYPE
     SYNTAX      InetAddress
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "IP address for gateway for transport stream if over IP.  If not
          used value shall be all zeros. May be IPv4 or IPv6 address depending
          on encoderCfgTransportInetAddrType."
     ::= { encoderCfgTransportEntry 12 }


encoderCfgTransportSubnetMask   OBJECT-TYPE
     SYNTAX      InetAddressPrefixLength
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Subnet mask normally used transport stream if over IP."
     ::= { encoderCfgTransportEntry 13 }


encoderCfgTransportIpFec   OBJECT-TYPE
     SYNTAX      INTEGER {
          off (1),
          on (2)
                }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Indicator showing whether forward error correction has
        been applied to the signal."
     ::= { encoderCfgTransportEntry 14 }


encoderCfgTransportIpInterleave   OBJECT-TYPE
     SYNTAX      INTEGER {
          off (1),
          on (2)
                }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Indicator showing whether interleaving has
        been applied to the signal."
     ::= { encoderCfgTransportEntry 15 }


encoderCfgTransportIpAddrgMode   OBJECT-TYPE
     SYNTAX      INTEGER {
          unicast (1),
          multicast (2),
          broadcast (3)     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Destination address type used if transport carried by IP."
     ::= { encoderCfgTransportEntry 16 }
```

```
-- ********************************************************************
-- ************************* INPUT MONITOR     ************************
-- ********************************************************************

encoderInputMonVideoTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF EncoderInputMonVideoEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
          "This table shows the status of the video input to this encoder."
      ::= { inputMonitor 1 }

encoderInputMonVideoEntry  OBJECT-TYPE
      SYNTAX      EncoderInputMonVideoEntry
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
          "An entry containing management information applicable to a
          particular encoder's input video.  This table is indexed by
          a unique IfIndex used for each video input supported by this
        agent. The individual encoder is identified by
        encoderInputMonVideoPhysIndex.
        Only the active input for each encoder is monitored. Disabled
        inputs shall not appear in this table."
      INDEX { encoderInputMonVideoIfIndex }
      ::= { encoderInputMonVideoTable 1 }

EncoderInputMonVideoEntry ::= SEQUENCE {
      encoderInputMonVideoIfIndex
          InterfaceIndex,
      encoderInputMonVideoChannelIndex
          Unsigned32,
      encoderInputMonVideoPhysIndex
          PhysicalIndex,
      encoderInputMonVideoType
          INTEGER,
      encoderInputMonVideoSyncLock
          INTEGER,
      encoderMonVideoInputFrameRate
          VideoInputFrameRateType,
      encoderInputMonVideoFrameLock
          INTEGER,
      encoderInputMonVideoChromaStable
          MpegErrorStatus,
      encoderInputMonVideoBlack
          INTEGER,
        encoderInputMonVideoLines
          INTEGER,
      encoderInputMonVideoSdiCk
          INTEGER
}

encoderInputMonVideoIfIndex  OBJECT-TYPE
      SYNTAX     InterfaceIndex
      MAX-ACCESS not-accessible
      STATUS     current
```

```
          DESCRIPTION
               "Identifier index for this video signal input."
          ::= { encoderInputMonVideoEntry 1 }



encoderInputMonVideoChannelIndex  OBJECT-TYPE
          SYNTAX      Unsigned32
          MAX-ACCESS read-only
          STATUS      current
          DESCRIPTION
               "Identifier for this video signal input when
            more than one video source present on the
            physical input interface."
          ::= { encoderInputMonVideoEntry 2 }



encoderInputMonVideoPhysIndex OBJECT-TYPE
          SYNTAX      PhysicalIndex
          MAX-ACCESS read-only
          STATUS      current
          DESCRIPTION
               "Cross reference to the entPhysicalIndex in the Entity MIB
            for the encoder processing this input."
          ::= { encoderInputMonVideoEntry 3 }



encoderInputMonVideoType  OBJECT-TYPE
          SYNTAX      INTEGER {
            other       (1),
               analogComposite (2),
               analogComponent (3),
               sdi (4),
               hdSdi (5),
            ip (6)
          }
          MAX-ACCESS read-only
          STATUS      current
          DESCRIPTION
               "Type of video interface"
          ::= { encoderInputMonVideoEntry 4 }



encoderInputMonVideoSyncLock  OBJECT-TYPE
          SYNTAX      INTEGER {
               locked (1),
               notLocked (2),
               intermittent (3)
          }
          MAX-ACCESS read-only
          STATUS      current
          DESCRIPTION
               "The encoder input is locked to the video sync.  Intermittent
               indicates at least one break in lock in since this object was last
               read."
          ::= { encoderInputMonVideoEntry 5 }
```

```
encoderMonVideoInputFrameRate  OBJECT-TYPE
     SYNTAX      VideoInputFrameRateType
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Frame rate as measured by the encoder."
     ::= { encoderInputMonVideoEntry 6 }


encoderInputMonVideoFrameLock  OBJECT-TYPE
     SYNTAX      INTEGER {
       notSupported(1),
          locked (2),
          notLocked (3),
          intermittent (4)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Indicates if the encoder is able to lock to the input
          frame rate."
     ::= { encoderInputMonVideoEntry 7 }


encoderInputMonVideoChromaStable  OBJECT-TYPE
     SYNTAX      MpegErrorStatus
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "The encoder is able to process the chroma information."
     ::= { encoderInputMonVideoEntry 8 }


encoderInputMonVideoBlack  OBJECT-TYPE
     SYNTAX      INTEGER {
          notSupported(1),
       good (2),
          black (3)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "The encoder detects no active video."
     ::= { encoderInputMonVideoEntry 9 }


encoderInputMonVideoLines  OBJECT-TYPE
     SYNTAX      INTEGER {
          notSupported(1),
       good (2),
          mismatch (3)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
        "The number of lines in the input video matches that configured
         in the encoder"
```

```
            ::= { encoderInputMonVideoEntry 10 }



encoderInputMonVideoSdiCk  OBJECT-TYPE
      SYNTAX    INTEGER {
            locked (1),
            notLocked (2),
            intermittent (3),
            notSdi (4)
      }
      MAX-ACCESS read-only
      STATUS    current
      DESCRIPTION
            "The encoder is locked to the SDI clock."
      ::= { encoderInputMonVideoEntry 11 }



encoderInputMonAudioTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF EncoderInputMonAudioEntry
      MAX-ACCESS not-accessible
      STATUS    current
      DESCRIPTION
            "This table shows the state of each audio input to this encoder."
      ::= { inputMonitor 2 }

encoderInputMonAudioEntry  OBJECT-TYPE
      SYNTAX     EncoderInputMonAudioEntry
      MAX-ACCESS not-accessible
      STATUS    current
      DESCRIPTION
            "An entry containing management information applicable to a
            particular encoder's input audio.  This table is indexed by the
            unique IfIndex used for each physical input supported by this
         agent, and encoderInputMonAudioInputChanIndex which identifies
         individual audio input channel when the interface carries more
         than one channel.
         encoderInputMonAudioPhysIndex  and encoderInputMonAudioCfgChannel
         are used together to identify the encoder and output channel
         carrying the audio from this input as defined in the
         encoderCfgATable.
         Only active inputs are monitored. Disabled/unused
         inputs shall not appear in this table."
      INDEX { encoderInputMonAudioIfIndex,
            encoderInputMonAudioInputChanIndex }
      ::= { encoderInputMonAudioTable 1 }

EncoderInputMonAudioEntry ::= SEQUENCE {
      encoderInputMonAudioIfIndex
            InterfaceIndex,
      encoderInputMonAudioInputChanIndex
            Unsigned32,
      encoderInputMonAudioPhysIndex
            PhysicalIndex,
      encoderInputMonAudioCfgChannel
            Unsigned32,
      encoderInputMonAudioType
            INTEGER,
```

```
        encoderInputMonAudioLevel
                Integer32,
        encoderInputMonAudioReference
                Integer32,
        encoderInputMonAudioSilence
                INTEGER,
        encoderInputMonAudioAesCk
                INTEGER,
        encoderInputMonAudioFraming
                INTEGER,
        encoderInputMonAudioAesType
                INTEGER
}

encoderInputMonAudioIfIndex  OBJECT-TYPE
        SYNTAX      InterfaceIndex
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
                "IF index used to identify the physical input for this audio
                component."
        ::= { encoderInputMonAudioEntry 1 }

encoderInputMonAudioInputChanIndex  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
                "Identifier index for this audio signal input when more than one
                channel shares one physical interface e.g. SMPTE-276 embedded
                audio."
        ::= { encoderInputMonAudioEntry 2 }


encoderInputMonAudioPhysIndex OBJECT-TYPE
        SYNTAX      PhysicalIndex
        MAX-ACCESS read-only
        STATUS      current
        DESCRIPTION
                "Cross reference to the entPhysicalIndex in the Entity MIB
           for the encoder processing this input."
        ::= { encoderInputMonAudioEntry 3 }


encoderInputMonAudioCfgChannel  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS not-accessible
        STATUS      current
        DESCRIPTION
                "Cross reference to identifier index for this audio channel.
                Directly corresponds to audio channel index in encoder audio
                configuration. Not valid if this input is disabled."
        ::= { encoderInputMonAudioEntry 4 }

--
encoderInputMonAudioType  OBJECT-TYPE
        SYNTAX      INTEGER {
```

```
               analog (1),
               aes (2),
               embedded (3)
          }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Type of this audio interface."
     ::= { encoderInputMonAudioEntry 5 }

encoderInputMonAudioLevel  OBJECT-TYPE
     SYNTAX      Integer32
     UNITS       "dB"
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "Average amplitude relative to 0dBm, 32767 indicates not
       supported."
     ::= { encoderInputMonAudioEntry 6 }


encoderInputMonAudioReference  OBJECT-TYPE
     SYNTAX      Integer32
     UNITS       "dB"
     MAX-ACCESS read-write
     STATUS      current
     DESCRIPTION
          "A field defining the nominal audio level
       for this input. If this information is provided
       by the encoder's configuration this field shall be
       read only. Otherwise this field may      be read-write to
       allow the monitoring system operator to store the figure
       for future reference. In this case the encoder shall
       provide non-volatile storage of the value but not
       act upon it."
     ::= { encoderInputMonAudioEntry 7 }


encoderInputMonAudioSilence  OBJECT-TYPE
     SYNTAX      INTEGER {
          audioPresent (1),
          silent (2),
          notSupported (3)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "No audio input detected for T seconds. T and threshold
       of silence configured using encoder's proprietary control."
     ::= { encoderInputMonAudioEntry 8 }

encoderInputMonAudioAesCk  OBJECT-TYPE
     SYNTAX      INTEGER {
          locked (1),
          notLocked (2),
          intermittent (3),
          notAes (4)
```

```
        }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "The encoder is locked to the AES clock or detects SMTE-272 data in
          SDI. Intermittent indicates currently good but loss of lock detected
       at least once since this object was last read."
     ::= { encoderInputMonAudioEntry 9 }

encoderInputMonAudioFraming  OBJECT-TYPE
     SYNTAX      INTEGER {
          locked (1),
          notLocked (2),
          intermittent (3),
          notAes (4)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "The correct framing of the digital audio input is recognized.
          Intermittent indicates at least one break in lock since this object
          was last read."
     ::= { encoderInputMonAudioEntry 10 }

encoderInputMonAudioAesType  OBJECT-TYPE
     SYNTAX      INTEGER {
          other (1),
          notRecognized (2),
          smpte337 (3),
          pcm (4)
     }
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
          "The type of data carried by the encoder input AES stream."
     ::= { encoderInputMonAudioEntry 11 }


encoderInputMonVbiTable  OBJECT-TYPE
     SYNTAX SEQUENCE OF EncoderInputMonVbiEntry
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
          "This table shows the status of each VBI line input to this
       encoder. Refer to encoderCfgVbiTable for VBI configuration.
       "
     ::= { inputMonitor 3 }

encoderInputMonVbiEntry  OBJECT-TYPE
     SYNTAX      EncoderInputMonVbiEntry
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
          "An entry containing management information applicable to a
          particular encoder's input VBI.  This table is indexed by the
          IfIndex of the video input carrying this VBI, an index for the
       VBI line itself and index for each field.
```

```
              encoderInputMonVbiPhysIndex is provided for identification of
              this encoder.
              Only the active input for each encoder is monitored. Disabled
              inputs shall not appear in this table."
          INDEX { encoderInputMonVbiIfIndex,
               encoderInputMonVbiLineIndex,
               encoderInputMonVbiFieldIndex }
         ::= { encoderInputMonVbiTable 1 }


EncoderInputMonVbiEntry ::= SEQUENCE {
     encoderInputMonVbiIfIndex
             InterfaceIndex,
    encoderInputMonVbiPhysIndex
             PhysicalIndex,
    encoderInputMonVbiLineIndex
             Unsigned32,
    encoderInputMonVbiFieldIndex
             Unsigned32,
    encoderInputMonVbiType
             INTEGER,
    encoderInputMonVbiErrors
             INTEGER,
      encoderInputMonVbiRate
             Integer32,
     encoderInputMonVbiTimeLastData
             TimeTicks
         }


encoderInputMonVbiIfIndex  OBJECT-TYPE
     SYNTAX      InterfaceIndex
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
             "IF index used to identify the physical input for this VBI
             service. This will normally have the same value as
        encoderInputMonVideoIfIndex"
         ::= { encoderInputMonVbiEntry 1 }



encoderInputMonVbiPhysIndex OBJECT-TYPE
     SYNTAX      PhysicalIndex
     MAX-ACCESS read-only
     STATUS      current
     DESCRIPTION
             "Cross reference to the entPhysicalIndex in the Entity MIB
          for the encoder processing this input."
         ::= { encoderInputMonVbiEntry 2 }



encoderInputMonVbiLineIndex  OBJECT-TYPE
     SYNTAX      Unsigned32 (6..23)
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
             "Identifier index for this VBI line input. Only values
             6..23 allowed. For PAL lines 318..335 corresponding
             values 6..23 are used with field index set to 2."
```

```
        ::= { encoderInputMonVbiEntry 3 }


encoderInputMonVbiFieldIndex  OBJECT-TYPE
      SYNTAX     Unsigned32 (1..3)
      MAX-ACCESS not-accessible
      STATUS     current
      DESCRIPTION
           "Identifier index for the field carrying this VBI line.
         Values 1 or 2 represent actual field number, 3 is provision
         for data being read from the same line on both fields."
      ::= { encoderInputMonVbiEntry 4 }


encoderInputMonVbiType  OBJECT-TYPE
      SYNTAX     INTEGER {
          correct (1),
          mismatch (2),
          notDiscriminated (3)     }
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "This object indicates whether the type of VBI matches
        the type configured for this line. Refer to
           encoderCfgVbiTable for VBI configuration."
      ::= { encoderInputMonVbiEntry 5 }



encoderInputMonVbiErrors  OBJECT-TYPE
      SYNTAX     INTEGER {
          good (1),
          errors (2),
          notChecked (3)     }
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "This object indicates whether the errors are detected
        in the VBI data on this line."
      ::= { encoderInputMonVbiEntry 6 }


encoderInputMonVbiRate  OBJECT-TYPE
      SYNTAX     Integer32
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
        "Rate of VBI data measured in bps. Where rate is too low
           for reporting a meaningful value this object shall be
           set to 0 and encoderInputMonVbiTimeLastData used. If
        measurement of the rate of this data is not supported
        by this encoder then -1 shall be returned."
      ::= { encoderInputMonVbiEntry 7 }



encoderInputMonVbiTimeLastData  OBJECT-TYPE
      SYNTAX     TimeTicks
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "Time in hundredths of a second since last data received
```

```
        on this line. A value approximating to the frame rate i.e.
        3 or 4 may be used to indicate continuous data is present.
            0 is reserved to indicate this measurement is currently
        not supported for this line.
        4,294,967,295 indicates never or beyond counter range. "
    ::= { encoderInputMonVbiEntry 8 }


encoderInputMonVancTable  OBJECT-TYPE
     SYNTAX SEQUENCE OF EncoderInputMonVancEntry
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
          "This table shows the status of each VANC service input to this
         encoder. Refer to encoderCfgVancTable for VANC configuration.
          "
     ::= { inputMonitor 4 }

encoderInputMonVancEntry  OBJECT-TYPE
     SYNTAX      EncoderInputMonVancEntry
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
          "An entry containing management information applicable to a
          particular encoder's input VANC.  This table is indexed by the
          IfIndex of the video input carrying this VANC and an index for
        the VANC service.
        encoderInputMonVancPhysIndex is provided for identification of
        this encoder.
        Only the active input for each encoder is monitored. Disabled
            inputs shall not appear in this table."
     INDEX { encoderInputMonVancIfIndex,
        encoderInputMonVancServiceIndex}
     ::= { encoderInputMonVancTable 1 }

EncoderInputMonVancEntry ::= SEQUENCE {
     encoderInputMonVancIfIndex
          InterfaceIndex,
   encoderInputMonVancPhysIndex
          PhysicalIndex,
   encoderInputMonVancServiceIndex
          Unsigned32,
   encoderInputMonVancType
          INTEGER,
   encoderInputMonVancErrors
          INTEGER,
   encoderInputMonVancRate
          Integer32,
   encoderInputMonVancTimeLastData
          TimeTicks
       }

encoderInputMonVancIfIndex  OBJECT-TYPE
     SYNTAX      InterfaceIndex
     MAX-ACCESS not-accessible
     STATUS      current
     DESCRIPTION
```

```
                  "IF index used to identify the physical input for this VANC
                  service. This will normally have the same value as
                  encoderInputMonVideoIfIndex "
         ::= { encoderInputMonVancEntry 1 }


encoderInputMonVancPhysIndex OBJECT-TYPE
     SYNTAX     PhysicalIndex
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Cross reference to the entPhysicalIndex in the Entity MIB
        for the encoder processing this input."
     ::= { encoderInputMonVancEntry 2 }


encoderInputMonVancServiceIndex  OBJECT-TYPE
     SYNTAX     Unsigned32 (6..23)
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
        "Identifier index for this VANC input.
        Has the same value as encoderCfgVancServiceIndex
        and provides cross reference between this table and
        encoderCfgVancTable."
     ::= { encoderInputMonVancEntry 3 }


encoderInputMonVancType  OBJECT-TYPE
     SYNTAX     INTEGER {
          correct (1),
          mismatch (2),
          notDiscriminated (3)     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "This object indicates whether the type of VANC matches
        the type configured for this line. Refer to
          encoderCfgVancTable for VANC configuration."
     ::= { encoderInputMonVancEntry 4 }


encoderInputMonVancErrors  OBJECT-TYPE
     SYNTAX     INTEGER {
          good (1),
          errors (2),
          notChecked (3)     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "This object indicates whether the errors are detected
        in the VANC data on this line."
     ::= { encoderInputMonVancEntry 5 }


encoderInputMonVancRate  OBJECT-TYPE
     SYNTAX     Integer32
```

```
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
        "Rate of Vanc data measured in bps. Where rate is too low
            for reporting a meaningful value this object shall be
            set to 0 and encoderInputMonVancTimeLastData used. If
        measurement of the rate of this data is not supported
        by this encoder then -1 shall be returned."
      ::= { encoderInputMonVancEntry 6 }



encoderInputMonVancTimeLastData  OBJECT-TYPE
      SYNTAX      TimeTicks
      MAX-ACCESS read-only
      STATUS      current
      DESCRIPTION
            "Time in hundredths of a second since last data received
          on this VANC service. A value approximating to the frame rate i.e.
          2, 3 or 4 may be used to indicate continuous data is present.
            0 is reserved to indicate this measurement is currently
          not supported for this Vanc service.
          4,294,967,295 indicates never or beyond counter range. "
      ::= { encoderInputMonVancEntry 7 }



encoderInputMonAncilTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF EncoderInputMonAncilEntry
      MAX-ACCESS not-accessible
      STATUS      current
      DESCRIPTION
            "This table shows the state of all other signal inputs to this
            encoder."
      ::= { inputMonitor 5 }

encoderInputMonAncilEntry  OBJECT-TYPE
      SYNTAX      EncoderInputMonAncilEntry
      MAX-ACCESS not-accessible
      STATUS      current
      DESCRIPTION
            "An entry containing management information applicable to a
            particular encoder's ancillary component input.  This table is
            indexed by a unique IfIndex used for each ancillary input
          supported by this agent, and encoderInputMonAncilChannelIndex
          which identifies individual ancillary input channel when the
          interface carries more than one channel.
          encoderInputMonAncilPhysIndex is provided for identification of
          this encoder.
          Only active inputs are monitored. Disabled/unused
          inputs shall not appear in this table."
      INDEX { encoderInputMonAncilIfIndex,
            encoderInputMonAncilChannelIndex }
      ::= { encoderInputMonAncilTable 1 }

EncoderInputMonAncilEntry ::= SEQUENCE {
      encoderInputMonAncilIfIndex
            InterfaceIndex,
```

```
        encoderInputMonAncilChannelIndex
              Unsigned32,
        encoderInputMonAncilPhysIndex
              PhysicalIndex,
        encoderInputMonAncilCfgChannel
              Unsigned32,
        encoderInputMonAncilType
              INTEGER,
        encoderInputMonAncilTimeLastPkt
              TimeTicks,
        encoderInputMonAncilRate
              Integer32,
        encoderInputMonAncilLock
              INTEGER,
        encoderInputMonAncilError
              MpegErrorStatus,
        encoderInputMonAncilHbMissed
              Counter32
            }

encoderInputMonAncilIfIndex  OBJECT-TYPE
     SYNTAX     InterfaceIndex
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
          "IF index used to identify the physical input for this ancillary
          component."
     ::= { encoderInputMonAncilEntry 1 }

encoderInputMonAncilChannelIndex  OBJECT-TYPE
     SYNTAX     Unsigned32
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
          "Identifier index for this ancillary signal input when more than one
          channel shares one physical interface e.g. IP input."
     ::= { encoderInputMonAncilEntry 2 }

encoderInputMonAncilPhysIndex OBJECT-TYPE
     SYNTAX     PhysicalIndex
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Cross reference to the entPhysicalIndex in the Entity MIB
        for the encoder processing this input."
     ::= { encoderInputMonAncilEntry 3 }


encoderInputMonAncilCfgChannel  OBJECT-TYPE
     SYNTAX     Unsigned32
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Cross reference to identifier index for this ancillary channel.
          Directly corresponds to ancillary component index in encoder
          ancillary configuration."
     ::= { encoderInputMonAncilEntry 4 }
```

```
encoderInputMonAncilType  OBJECT-TYPE
     SYNTAX     INTEGER {
          other (1),
          dsmcc (2),
          scte104Ip (3),
          scte104Vanc (4),
          subtitles (5),
          codeDownload (6),
          smpte333 (7),
          vbiData (8),
          vancData (9)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Type of this ancillary interface."
     ::= { encoderInputMonAncilEntry 5 }

encoderInputMonAncilTimeLastPkt  OBJECT-TYPE
     SYNTAX     TimeTicks
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Time in hundredths of a second since last packet received, 0
          indicates less than 0.01 second, 4,294,967,295 indicates never or
        beyond counter range. Only applicable to very slow or intermittent
        data, other wise encoderInputMonAncilRate should be used. "
     ::= { encoderInputMonAncilEntry 6 }

encoderInputMonAncilRate  OBJECT-TYPE
     SYNTAX     Integer32
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Number of bits received within timing period. May be averaged if
        result is clearer. -1 indicates not monitored."
     ::= { encoderInputMonAncilEntry 7 }


encoderInputMonAncilLock  OBJECT-TYPE
     SYNTAX     INTEGER {
          good (1),
          notLocked (2),
          intermittent (3),
          readyToAcquire (4)
     }
     MAX-ACCESS read-only
     STATUS     current
     DESCRIPTION
          "Encoder locked to data stream framing. Ready to acquire
          indicates correct response of encoder during gaps in data."
     ::= { encoderInputMonAncilEntry 8 }

encoderInputMonAncilError  OBJECT-TYPE
     SYNTAX     MpegErrorStatus
```

```
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "Error detected in this ancillary stream since this object was last
           read."
      ::= { encoderInputMonAncilEntry 9 }

encoderInputMonAncilHbMissed  OBJECT-TYPE
      SYNTAX     Counter32
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
           "Number of heartbeats missed since last heartbeat successfully
           received."
      ::= { encoderInputMonAncilEntry 10 }


-- ********************************************************************
-- ***********************  COMPLIANCE     ************************
-- ********************************************************************



mpegEncoderCompliance  MODULE-COMPLIANCE
      STATUS     current
      DESCRIPTION
           "The minimum compliance statement for MPEG encoders."
      MODULE
           GROUP encoderCfgGroup
           DESCRIPTION
                "The encoderCfgGroup is unconditionally optional."
           GROUP encoderInputGroup
           DESCRIPTION
                "The encoderInputGroup is unconditionally optional."
           ::= { mpegEncoderMIBCompliances 1 }


encoderCfgGroup  OBJECT-GROUP
      OBJECTS { encoderCfgAncilEnabled,
           encoderCfgAncilServiceIndex,
           encoderCfgAncilLanguage,
           encoderCfgAncilPcrPid,
           encoderCfgAncilPid,
           encoderCfgAncilRate,
           encoderCfgAncilSourceType,
           encoderCfgAncilType,
           encoderCfgAudioDialogNorm,
           encoderCfgAudioEnabled,
           encoderCfgAudioInputType,
           encoderCfgAudioLanguageA,
           encoderCfgAudioLanguageB,
           encoderCfgAudioLipSync,
           encoderCfgAudioMode,
           encoderCfgAudioPcrPid,
           encoderCfgAudioPid,
           encoderCfgAudioRate,
           encoderCfgAudioPassThru,
```

```
                encoderCfgAudioInputLabel,
                encoderCfgAudioStandard,
                encoderCfgAudioServiceIndex,
                encoderCfgServiceId,
                encoderCfgServiceName,
                encoderCfgServiceTransportIndex,
                encoderCfgServiceType,
                encoderCfgTransportIpFec,
            encoderCfgTransportIpInterleave,
                encoderCfgTransportIpAddrgMode,
                encoderCfgTransportDestIpAddr,
                encoderCfgTransportDestUdpPort,
                encoderCfgTransportEnabled,
                encoderCfgTransportGatewayAddr,
                encoderCfgTransportId,
                encoderCfgTransportInetAddrType,
                encoderCfgTransportRate,
                encoderCfgTransportRateMode,
                encoderCfgTransportSiStd,
                encoderCfgTransportSubnetMask,
                encoderCfgTransportType,
                encoderCfgTransportIfIndex,
            encoderCfgVancDid,
            encoderCfgVancSdid,
            encoderCfgVancType,
            encoderCfgVancCarriage,
            encoderCfgVancCompIndex,
                encoderCfgVbiField,
            encoderCfgVbiLine,
            encoderCfgVbiType,
            encoderCfgVbiCarriage,
            encoderCfgVbiCompIndex,
            encoderCfgVideoInputIf,
            encoderCfgVideoInputFrameRate,
            encoderCfgVideoInputScan,
            encoderCfgVideoInputFormat,
                encoderCfgVideoAspectRatio,
                encoderCfgVideoBitrateAvg,
            encoderCfgVideoBitrateMax,
            encoderCfgVideoBitrateMin,
                encoderCfgVideoBorderProcessing,
                encoderCfgVideoCodingDelay,
                encoderCfgVideoDeblockAlpha,
                encoderCfgVideoDeblockBeta,
                encoderCfgVideoDeblockEnable,
                encoderCfgVideoFilmMode,
                encoderCfgVideoIdrRate,
                encoderCfgVideoPcrPid,
                encoderCfgVideoPesAlignment,
                encoderCfgVideoPid,
                encoderCfgVideoHorzResolution,
            encoderCfgVideoVertResolution,
            encoderCfgVideoServiceIndex,
            encoderCfgVideoType,
                encoderCfgVideoCompression,
                encoderCfgVideoRateMode }
        STATUS      current
```

```
        DESCRIPTION
            "A collection of objects that provide information applicable to a
            particular encoder's configuration parameters."
        ::= { mpegEncoderMIBGroups 1 }


encoderInputGroup  OBJECT-GROUP
        OBJECTS {   encoderInputMonAncilError,
            encoderInputMonAncilHbMissed,
            encoderInputMonAncilLock,
            encoderInputMonAncilRate,
            encoderInputMonAncilCfgChannel,
            encoderInputMonAncilTimeLastPkt,
            encoderInputMonAncilType,
          encoderInputMonVbiType,
        encoderInputMonVbiErrors,
        encoderInputMonVbiRate,
        encoderInputMonVbiTimeLastData,
            encoderInputMonVancType,
        encoderInputMonVancErrors,
        encoderInputMonVancRate,
        encoderInputMonVancTimeLastData,
            encoderInputMonAudioAesCk,
            encoderInputMonAudioAesType,
            encoderInputMonAudioFraming,
            encoderInputMonAudioReference,
            encoderInputMonAudioLevel,
            encoderInputMonAudioSilence,
            encoderInputMonAudioType,
            encoderInputMonVideoBlack,
            encoderInputMonVideoChromaStable,
            encoderInputMonVideoLines,
            encoderMonVideoInputFrameRate,
            encoderInputMonVideoFrameLock,
            encoderInputMonVideoSdiCk,
            encoderInputMonVideoSyncLock,
            encoderInputMonVideoType,
            encoderInputMonVideoChannelIndex,
            encoderInputMonVideoPhysIndex,
            encoderInputMonAudioPhysIndex,
            encoderInputMonVbiPhysIndex,
            encoderInputMonVancPhysIndex,
            encoderInputMonAncilPhysIndex
            }
        STATUS      current
        DESCRIPTION
            "A collection of objects that provide information applicable to a
            particular encoder's input parameters."
        ::= { mpegEncoderMIBGroups 2 }


END
```