Video Services Forum (VSF)
Technical Recommendation TR-10-3

Internet Protocol Media Experience (IPMX):
PCM Digital Audio

April 25, 2023

DRAFT

INTELLECTUAL PROPERTY RIGHTS

RECIPIENTS OF THIS DOCUMENT ARE REQUESTED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT CLAIMS OR OTHER INTELLECTUAL PROPERTY RIGHTS OF WHICH THEY MAY BE AWARE THAT MIGHT BE INFRINGED BY ANY IMPLEMENTATION OF THE RECOMMENDATION SET FORTH IN THIS DOCUMENT, AND TO PROVIDE SUPPORTING DOCUMENTATION.

THIS RECOMMENDATION IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NONINFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS RECOMMENDATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE FORUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS RECOMMENDATION.

LIMITATION OF LIABILITY

VSF SHALL NOT BE LIABLE FOR ANY AND ALL DAMAGES, DIRECT OR INDIRECT, ARISING FROM OR RELATING TO ANY USE OF THE CONTENTS CONTAINED HEREIN, INCLUDING WITHOUT LIMITATION ANY AND ALL INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS, LOSS OF PROFITS, LITIGATION, OR THE LIKE), WHETHER BASED UPON BREACH OF CONTRACT, BREACH OF WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE FOREGOING NEGATION OF DAMAGES IS A FUNDAMENTAL ELEMENT OF THE USE OF THE CONTENTS HEREOF, AND THESE CONTENTS WOULD NOT BE PUBLISHED BY VSF WITHOUT SUCH LIMITATIONS.

Executive Summary

Internet Protocol Media Experience (IPMX) was created to foster the adoption of open standards-based protocols for interoperability over IP in the media and entertainment (M&E) and professional audio/video industries. IPMX is based on the SMPTE ST 2110 and as such the VSF TR-10 suite of Technical Recommendations is built as set of differences between SMPTE ST 2110 and IPMX.

This Technical Recommendation corresponds to the SMPTE ST 2110-30 document and describes the transport of PCM digital audio using RTP protocol in IPMX. It documents the differences between TR-10-3 and SMPTE ST 2110-30. Some of the subject covered in this document include the payload format, sample rate, Media Clock, RTP Clock, RTP Timestamps and the IPMX Info Block definition for PCM digital audio.

Table of Contents

1 Introduction (Informative)

IPMX, which stands for IP Media Experience, is based on two families of specifications. The SMPTE ST 2110 Professional Media Over Managed IP Networks suite of standards for the transport of video, audio, and ancillary/control signals over IP networks, and the NMOS REST APIs from AMWA, which provide discovery, connection management, and control.

IPMX is an accessible, open standard that meets the needs of professional and consumer video and audio users in a wide variety of contexts while giving manufacturers and developers what they need to build low-latency, interoperable, IP based audiovisual products or applications.

This document covers the IPMX transport of PCM Digital Audio using the RTP protocol. Other parts of the TR-10 family of Technical Recommendation (TRs) describe IPMX individual media essence types, along with their requirements, and defines other aspects of the IPMX system.

2 Contributors

The following individuals participated in the Video Services Forum IPMX working group that developed this Technical Recommendation.

Aaron Doughten (Sencore)	Charles Buysschaert (Intopix)	Jean-Baptiste Lorent (IntoPIX)	Phil Nguyen (Nextera)
Alain Bouchard (Matrox)	Chris Lapp (Cisco)	Jed Deame (Nextera Video)	Prinyar Boon (Phabrix)
Albert Faust (Arista)	Clark Williams (Christie Digital)	JJ Eynon (CNN)	Raul Diaz (Intel)
Andre Testa (Matrox)	Daniel BOUQUET (Analogway)	John Belstner (Intel)	Raymond Hermans (Adeas)
Andreas Hildebrand (ALC NetworX)	Danny Pierini (Matrox)	John Dale (Media Links)	Robert Welch (Arista)
Andrew Starks (Macnica)	David Chiappini (Matrox)	John Fletcher (BBC)	Ron Stites (Macnica)
Antoine Hermans (Adeas)	David Mitchinson (Appear TV)	Karl Johnson (Christie Digital)	Tadahiro Watanabe (Macnica)
Arnaud Germain (Intopix)	Gerard Phillips (Arista)	Karl Paulsen (Diversified)	Teiji Kubota (Macnica)
Ben Cope (Intel)	Greg Schlechter (Intel)	Marc Levy (Macnica)	Thomas True (NVIDIA)
Brad Gilmer (VSF)	Greg Stigall (Warner Media)	Mike Boucke (AJA)	Tim Bruylants (intopix)
Bob Ruhl (VSF)	Jack Douglass (PacketStorm)	Paulo Francisco (Evertz)	Wes Simpson (LearnIPvideo)
Cassidy Phillips (Imagine)	Jean Lapierre (Matrox)	Peter Brightwell (BBC)	

3 About the Video Services Forum

The Video Services Forum, Inc. (www.videoservicesforum.org) is an international association dedicated to video transport technologies, interoperability, quality metrics and education. The VSF is composed of service providers, users and manufacturers. The organization's activities include:

- providing forums to identify issues involving the development, engineering, installation, testing and maintenance of audio and video services;
- exchanging non-proprietary information to promote the development of video transport service technology and to foster resolution of issues common to the video services industry;
- identification of video services applications and educational services utilizing video transport services;
- promoting interoperability and encouraging technical standards for national and international standards bodies.

The VSF is an association incorporated under the Not For Profit Corporation Law of the State of New York. Membership is open to businesses, public sector organizations and individuals worldwide. For more information on the Video Services Forum or this document, please call +1 929-279-1995 or e-mailopsmgr@videoservicesforum.org.

4 Conformance Notation

Normative text describes elements of the design that are indispensable or contain the conformance language keywords: "shall," "should," or "may."

Informative text is potentially helpful to the user but not indispensable and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except the Introduction and any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:"

The keywords "shall" and "shall not" indicate requirements strictly to be followed to conform to the document and from which no deviation is permitted.

The keywords "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword "reserved" indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified, the order of precedence of the types of normative information in this document shall be as follows: Normative prose shall be the authoritative definition; Tables shall be next; followed by formal languages; then figures; and then any other language forms.

5 Normative References

- SMPTE ST 2110-30:2017 Professional Media Over Managed IP Networks: PCM Digital Audio
- SMPTE ST 2110-10:2022 Professional Media over Managed IP Networks: System Timing and Definitions
- VSF TR-10-1 Internet Protocol Media Experience (IPMX): System Timing and Definitions
- AES67-2018 AES standard for audio applications of networks - High-performance streaming audio-over-IP interoperability
- Internet Engineering Task Force (IETF) RFC 4566 SDP: Session Description Protocol
- Internet Engineering Task Force (IETF) RFC 3551 RTP Profile for Audio and Video Conferences with Minimal Control
- Internet Engineering Task Force (IETF) RFC 3190 RTP Payload Format for 12-bit DAT Audio and 20- and 24-bit Linear Sampled Audio

6 Definitions

For the purposes of this document, the terms, and definitions of VSF TR-10-1 and those of SMPTE ST 2110-30 apply.

7 General Provisions

All audio PCM IPMX Senders and Receivers compliant with this TR shall comply with the following specifications:

SMPTE ST 2110-30 Sections 1-5 and 6.2.2

IPMX network interface requirements shall be in accordance with the provisions of SMPTE ST 2110-10 section 6, subject to the additional constraints in this document.

All IPMX Media streams shall have a UDP destination port value that is even, and that is greater than 1024.

All IPMX Media streams should have a UDP destination port value that is greater than 5000.

```
Note: The interested reader can refer to RFC 3551 section 8 for
a description of the selection of the above port number range.
```

Audio PCM IPMX Sender's digital audio streams shall conform to AES67, including the Session Description Protocol (SDP) as described in IETF RFC 4566, subject to the constraints in this document and those of TR-10-1.

Notwithstanding the provisions regarding SIP in AES67, audio PCM IPMX Receivers need not support Session Initiation Protocol (SIP) for audio nor any other specific method mentioned in AES67 for audio connection management.

IPMX Senders shall make their SDP object available through the management programming interface of the device.

The UDP size of each RTP packet shall not exceed the Standard UDP Size Limit as specified in SMPTE ST 2110-10.

8 Payload Formats and Sample Rates

IPMX Senders and Receivers of PCM digital audio shall support a digital audio sample rate of 48 kHz.

IPMX Senders and Receivers of PCM digital audio should support digital audio sample rates of 44.1 kHz and 96 kHz.

IPMX PCM digital audio supports the following payload encoding formats:

L16 16-bit linear format defined in RFC 3551 clause 4.5.11

L24 24-bit linear format defined in RFC 3190 clause 4

When operating at 48 kHz sampling rate, IPMX Receivers shall support both L16 and L24 encodings.

When operating at 48 kHz sampling rate, IPMX Senders shall support either L16 or L24 encoding and may support both encodings.

When operating at 44.1 kHz sampling rate, IPMX Senders and Receivers shall support L16 encoding.

When operating at 96 kHz sampling rate, IPMX Senders and Receivers shall support L24 encoding.

Devices may support other audio format and sampling rate combinations. However, using combinations of sampling rates and payloads encoding beyond those defined in this document are outside the scope of this TR.

9 Media Clock, RTP Clock, and RTP Timestamps

The Media Clock, RTP Clock and RTP Timestamps shall comply with the provisions of VSF TR-10-1.

The rate of the Media Clock and RTP Clock shall be the same as the digital audio sample rate.

10 IPMX Sender and Receiver Conformance Levels

Audio IPMX Receivers shall support reception of any audio stream that conforms to Conformance Level A as set forth in SMPTE ST 2110-30, section 7, irrespective of the number of audio channels the IPMX Receiver can process.

Audio IPMX Senders shall support transmission of at least one audio stream containing a number of audio channels that can be received by an IPMX Receiver conforming to this TR.

IPMX Receivers may support other conformance levels described in SMPTE ST 2110-30 section 7 table 2.

11 IPMX Info Block definition for PCM digital audio

IPMX Senders shall send RTCP Sender Reports as outlined in TR-10-1. These RTCP Sender Reports shall include an IPMX Info Block and a Media Info Block.

The format of a IPMX Info Block extension including a IPMX PCM digital audio Media Info Block shall be as in Figure 1 below.

```
         0                   1                   2                   3
         0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
IPMX    |       IPMX tag = 0x5831       |    IPMX Info Block length     |
Info    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Block   | block version |                  reserved                     |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                       ts-refclk String                        |
        :                             . . .                             :
        |                                                               |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                        mediaclk string                        |
        |                                                               |
        |                                                               |
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
Media   | Media Info Block type = 0x0002|   Media Info Block length     |
Info    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Block   |                         sampling rate                         |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |sample Size    |channel Count  |        packet time            |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                      measuredsamplerate                       |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
|                        channel-order string length                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          channel-order string                         |
|                                  . . .                                |
|                                                                       |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Figure 1 – IPMX header extension for PCM digital audio media

IPMX tag: 16 bits
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

IPMX Info Block length: 16 bits
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

block version: 8 bits
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

reserve: 24 bits
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

ts-refclk string: 64 bytes
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

mediaclk string: 12 bytes
See TR-10-1 section 8.7 (RTCP Sender Report General Provision)

Media Info Block type: 16 bits
Shall contain the constant 0x0002. This identifies the Media Info Block for PCM Digital Audio.

Media Info Block length: 16 bits
The length of the Media Info Block in 32-bit words minus one, including the header, the Media Info Block content and any padding required to align the Media Info Block on a 32 bits boundary.

sampling rate: 32 bits
Shall correspond to the sampling rate in Hz used for sample encodings in the RTP payload as defined in AES67 section 7.1

sample size: 8 bits
Shall correspond to the number of bits used for sample encodings in the RTP payload as defined in AES67 section 7.1, value shall be either 16 or 24.

channel count: 8 bits
Shall be equal to the number of channels present in the RTP PCM audio stream.

packet time: 16 bits
Shall be equal to the packet time in microseconds corresponding to the ptime attribute defined in AES67 section 8.1

measuredsamplerate : 32 Bits
Shall correspond to the measured sample rate parameter in Hz as defined in TR-10-1 section 10.3 (Baseband Audio Sender Signaling) for the current PCM audio stream.

channel-order string length: 32 bits
Shall be the length of the channel-order string in 32-bit words, including any padding

channel-order string: variable size
Shall correspond to the channel-order string value as defined in SMPTE ST 2110-30 section 6.2.2 padded with 0x0 byte values to align on a 32-bit boundary

12 RTCP Sender Report Example (Informative)

The following shows an example of a RTCP Sender Report for an IPMX Audio Sender that has a synchronization source identifier (SSRC) value of 2345, sending the RTCP Sender Report for RTP timestamp of 4070650991 after having sent 9000560 packets whose SDP file content has changed 3 times since it was activated and having the following SDP file:

```
v=0
o=- 1618348182647029200 1618348303876470900 IN IP4 25.25.30.151
s=IP audio OUT 1
t=0 0
m=audio 10000 RTP/AVP 97
c=IN IP4 239.30.0.1/128
a=source-filter: incl IN IP4 239.30.0.1 25.25.30.151
a=rtpmap:97 L24/48000/8
a=fmtp:97 channel-order=SMPTE2110.(U08); IPMX;
measuredsamplerate=47952
a=ts-refclk:localmac=00-20-FC-32-2F-40
a=ptime:0.12
a=mediaclk:sender
```

Referring to Figure 1 from TR-10-1 the value for the different fields of the IPMX RTCP Sender Report corresponding to this example would be:

V = 2
P = 0
RC = 0
PT = 200
length = 36

> Note: 36 is the size in 32-bit words of the packet payload - 1 as specified in RFC 3550 section 6.4.1. It is obtained by summing the length in Bytes of RTCP Sender report header (8) plus the length of the sender info (20) plus the length of the IPMX Info Block (2 bytes for

IMPX tag, 2 for IPMX length, 1 for the IPMX Info Block version counter + 3 reserved bytes + 64 for ts-refclk string + 12 for mediaclk string) + the length of the Media Info Block (2 for the media block type + 2 for the media info length + 4 for the sampling rate + 1 for the sample size + 1 for the channel count + 2 for the packet time + 4 for the measuredsamplerate + 4 for the channel-order string length + the actual length of the channel-order string field). Thus, the total size in Bytes is 8+20+(2+2+1+3+64+12) + (2+2+4+1+1+2+4+4+16) = 148. To get the amount in 32-bit words we divide by 4 and get 37. And finally subtract 1 to get 36.

SSRC = 2345
NTP timestamp, most significant word = 1666377592
NTP timestamp, least significant word = 777737730
RTP timestamp = 4070650991
sender’s packet count = 9000560
sender's octet count = 432026880
IPMX tag = 0x5831
IPMX Info Block length = 29

Note: 29 is the size in 32-bit words of the IPMX Info Block – 1 as specified in section 11 of this document. It is obtained by summing the length in Bytes as follows (2 for IMPX tag, 2 for IPMX Info Block length, 1 for the IPMX Info Block version counter + 3 reserved bytes + 64 for ts-refclk string + 12 for mediaclk string) + the length of the Media Info Block (2 for the Media Info Block type + 2 for the Media Info Block length + 4 for the sampling rate + 1 for the sample size + 1 for the channel count + 2 for the packet time + 4 for the measuredsamplerate + 4 for the channel-order string length + the actual length of the channel-order string field). Thus, the total size in Bytes is (2+2+1+3+64+12) + (2+2+4+1+1+2+4+4+16) = 120. To get the amount in 32bit words we divide by 4 and get 30. And finally subtract 1 to get 29.

IPMX Info Block version counter = 3
ts-refclk string = “localmac=00-20-FC-32-2F-40”
mediaclk string =”sender”
Media Info Block type = 0x0002
Media Info Block length: 8

Note: 8 is the size in 32-bit words of the Media Info Block – 1 as specified in section **Error! Reference source not found.** of this document. It is obtained by summing the length in Bytes as follows 2 for the Media Info Block type + 2 for the Media Info Block length + 4 for the sampling rate + 1 for the sample size + 1 for the channel count + 2 for the packet time + 4 for the measuredsamplerate + 4 for the channel-order string length + the actual length of the channel-order string field). Thus, the total size in Bytes is (2+2+4+1+1+2+4+4+16) = 36. To get the amount in 32bit words we divide by 4 and get 9. And finally subtract 1 to get 8.

sampling rate=48000
sample size=24
channel count=8

packet time=125
measuredsamplerate =47952
channel-order string length=4

```
Note: 4 is the size in 32bit words of the channel-order string as
specified in section 11 of this document. It is obtained by taking the
length of the channel-order string field in bytes (including the 0x0
byte padding) and diving by 4.
```

channel-order string=”SMPTE2110.(U08)”

The following figure shows the corresponding RTCP Sender Report data.

```
         0                   1                   2                   3
         0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
header  |  2|0|     0     |       200     |              36               |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                             2345                              |
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
sender  |                           1666377592                          |
info    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                           777737730                           |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                           4070650991                          |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                            9000560                            |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                           432026880                           |
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
IPMX    |             0x5831            |              29               |
info    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
block   |        3      |                       0                       |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                      0x6c 0x6f 0x63 0x61                      |
        |                      0x6c 0x6d 0x61 0x63                      |
        |                      0x3d 0x30 0x30 0x2d                      |
        |                      0x32 0x30 0x2d 0x46                      |
        |                      0x43 0x2d 0x33 0x32                      |
        |                      0x2d 0x32 0x46 0x2d                      |
        |                      0x34 0x30 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                      0x73 0x65 0x6e 0x64                      |
        |                      0x65 0x72 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        |                      0x00 0x00 0x00 0x00                      |
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
media   |            0x0002             |               8               |
```

```
Info    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
block   |                             48000                             |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |      24       |       8       |              125              |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                             47952                             |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                               4                               |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                     0x53 0x4d 0x50 0x54                       |
        |                     0x45 0x32 0x31 0x31                       |
        |                     0x30 0x2e 0x28 0x55                       |
        |                     0x30 0x38 0x29 0x00                       |
        +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Figure 2 – PCM digital audio IPMX RTCP Sender Report example