

明 細 書

発明の名称：送信装置、送信方法、受信装置および受信方法

技術分野

[0001] 本技術は、送信装置、送信方法、受信装置および受信方法に関し、詳しくは、ビデオ、オーディオなどの伝送メディアをRF通信路や通信ネットワーク通信路を通じて送信する送信装置等に関する。

背景技術

[0002] 従来、例えば、特許文献1に記載されるように、放送向けのトランスポート構造としてMPEG2-TSが使用されている。

先行技術文献

特許文献

[0003] 特許文献1：特開2011-217161号公報

発明の概要

発明が解決しようとする課題

[0004] 近年、次世代放送向けのトランスポート構造としてMMT (MPEG Media Transport) 構造が脚光を浴びつつある。このMMT構造は、IPネットワークとの共存が主な特徴である。

[0005] 本技術の目的は、次世代放送向けのトランスポート構造において、デコード時刻および／または表示時刻を取得するための時刻情報を受信側に良好に伝送可能とすることにある。

課題を解決するための手段

[0006] 本技術の概念は、

ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを生成する伝送ストリーム生成部と、

上記伝送ストリームを所定の伝送路を通じて受信側に送信する伝送ストリーム送信部と、

上記第1の伝送パケットまたは上記第2の伝送パケットに、上記受信側でデコード時刻および/または表示時刻を取得するための時刻情報を挿入する時刻情報挿入部とを備える

送信装置にある。

[0007] 本技術において、伝送ストリーム生成部により、伝送ストリームが生成される。この伝送ストリームは、ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化されたものである。例えば、第1の伝送パケットおよび第2の伝送パケットはMMTパケットである、ようにされてもよい。伝送ストリーム送信部により、伝送ストリームは、所定の伝送路を通じて、受信側に送信される。例えば、所定の伝送路は、RF伝送路または通信ネットワーク伝送路である、ようにされてもよい。

[0008] 時刻情報挿入部により、第1の伝送パケットまたは第2の伝送パケットに、受信側でデコード時刻および/または表示時刻を取得するための時刻情報が挿入される。例えば、第1の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、時刻情報挿入部で挿入される時刻情報は、一つ以上のアクセスユニットにそれぞれ対応したデコード時刻および/または表示時刻を取得するための情報である、ようにされてもよい。

[0009] この場合、時刻情報挿入部で挿入される時刻情報は、一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までのオフセット値とを有する、ようにされてもよい。このようにオフセット値を用いることで、時刻情報の伝送を効率よく行うことが可能となる。また、この場合、時刻情報挿入部で挿入される時刻情報は、一つ以上のアクセスユニットのそれぞれに対応した表示時刻、あるいは表示時刻およびデコード時刻である、ようにされてもよい。このように表示時刻、あるいは表示時刻およびデコード時刻そのものを伝送することで、受信側における処理負荷の軽減が可能となる。

[0010] そして、この場合、オフセット値は、絶対オフセット値に対応した相対オフセット値であり、時刻情報挿入部で挿入される時刻情報には、相対オフセット値を絶対オフセット値に変換するための変換情報が付加される、ようにされてもよい。このように相対オフセット値を用いることで、オフセット値を受信側に効率よく伝送することが可能となる。例えば、時刻情報挿入部は、相対オフセット値を可変長符号化した後に挿入する、ようにされてもよい。このように可変長符号化を利用することで、時刻情報の伝送容量の低減を図ることが可能となる。

[0011] 例えば、第1の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、時刻情報挿入部は、ペイロードヘッダ部に、時刻情報を挿入する、ようにされてもよい。また、例えば、第1の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、ペイロード本体部には、伝送メディアがフラグメント化された一つ以上のアクセスユニットをそれぞれ含むフラグメントペイロードがフラグメントヘッダを伴って配置され、時刻情報挿入部は、フラグメントヘッダあるいはフラグメントペイロードに、対応するアクセスユニットの時刻情報を挿入する、ようにされてもよい。また、例えば、第2の伝送パケットのペイロード部は、ペイロードヘッダ部とペイロード本体部とからなり、時刻情報挿入部は、ペイロード本体部に、時刻情報を挿入する、ようにされてもよい。

[0012] このように本技術においては、第1の伝送パケットまたは第2の伝送パケットに、受信側でデコード時刻および表示時刻を取得するための時刻情報が挿入されるものであり、次世代放送向けのトランスポート構造において、デコード時刻および／または表示時刻を取得するための時刻情報を受信側に良好に伝送可能となる。

[0013] また、本技術の他の概念は、

ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを、送信側から所定の伝送路を通じて受信する伝送

ストリーム受信部を備え、

上記第 1 の伝送パケットまたは上記第 2 の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、

上記伝送ストリームから取り出された上記伝送メディアを、上記時刻情報に基づいて取得された上記デコード時刻および/または上記表示時刻を用いて処理する伝送メディア処理部をさらに備える

受信装置にある。

[0014] 本技術において、伝送ストリーム受信部により、所定の伝送路を通じて、送信側から伝送ストリームが受信される。この伝送ストリームは、ペイロードに伝送メディアを含む第 1 の伝送パケットと、ペイロードに伝送メディアに関する情報を含む第 2 の伝送パケットとが時分割的に多重化されたものである。第 1 の伝送パケットまたは第 2 の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されている。伝送メディア処理部により、伝送ストリームから取り出された伝送メディアが、時刻情報に基づいて取得されたデコード時刻および/または表示時刻を用いて処理される。

[0015] 例えば、第 1 の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、時刻情報は、一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までの絶対オフセット値に対応した相対オフセット値であり、この相対オフセット値を絶対オフセット値に変換するオフセット値変換部をさらに備える、ようにされてもよい。

[0016] このように本技術においては、第 1 の伝送パケットまたは第 2 の伝送パケットにデコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、この時刻情報に基づいてデコード時刻および/または表示時刻を取得し、伝送ストリームから取り出された伝送メディアの処理を良好に行うことができる。

発明の効果

[0017] 本技術によれば、次世代放送向けのトランスポート構造において、デコード時刻および／または表示時刻を取得するための時刻情報を受信側に良好に伝送できる。なお、本明細書に記載された効果はあくまで例示であって限定されるものではなく、また付加的な効果があってもよい。

図面の簡単な説明

- [0018] [図1]実施の形態としての送受信システムの構成例を示すブロック図である。
- [図2]MMTペイロード（MMT payload）の構成を概略的に示す図である。
- [図3]MMTファイルと実際に伝送されるMMTパケットとの対応関係の一例を示す図である。
- [図4]MMTファイルと実際に伝送されるMMTパケットとの対応関係の他の一例を示す図である。
- [図5]MMTファイルと実際に伝送されるMMTパケットとの対応関係のさらに他の一例を示す図である。
- [図6]フラグメント化されたMFUを含むMMTパケットを伝送パケットとする場合であって、そのMMTパケットがランダムアクセスの先頭（GOPの先頭）のパケットである場合を説明するための図である。
- [図7]フラグメント化されたMFUを含むMMTパケットを伝送パケットとする場合であって、そのMMTパケットがランダムアクセスの非先頭のパケットである場合を説明するための図である。
- [図8]フラグメント化されたMFUを含むMMTパケットを伝送パケットとする場合であって、そのMMTパケットがランダムアクセスの先頭（GOPの先頭）のパケットである場合を説明するための図である。
- [図9]MMTパケットの構成をツリー形式で示す図である。
- [図10]トランスポートパケット送信装置およびトランスポートパケット受信装置の概念図である。
- [図11]「NTP shortフォーマット」を示す図である。
- [図12]メディアデータがビデオであり、初期値がデコード時刻である場合に

おける時刻情報を説明するための図である。

[図13]メディアデータがビデオであり、初期値が表示時刻である場合における時刻情報を説明するための図である。

[図14]メディアデータがオーディオであり、初期値がデコード時刻である場合における時刻情報を説明するための図である。

[図15]デコード・出力処理部の構成例を示すブロック図である。

[図16]制御部におけるAV同期再生制御の一例を示すタイミングチャートである。

[図17]MFUにフラグメントされているパケットが伝送上でドロップされた場合について説明するための図である。

[図18]MMTパケットに時刻情報を挿入する方法の一例を概略的に示す図である。

[図19]送受信遅延を低くするために、各アクセスユニットに対応して時刻情報（タイムスタンプ）を挿入する場合のパケット構成の一例を示す図である。

[図20]MMTパケットの全体の構造例を示す図である。

[図21]MMTパケットヘッダ（`mmtp_header()`）の構造例を示す図である。

[図22]MMTペイロードヘッダ（`mmtp_payload_header()`）の構造例を示す図である。

[図23]MMTペイロード・ヘッダ・エクステンション（`mmtp_payload_header_extension()`）の構造例を示す図である。

[図24]MMTペイロード・ヘッダ・エクステンションの構造例の主要な情報の内容を示す図である。

[図25]MMTペイロード・ヘッダ・エクステンション（`mmtp_payload_header_extension()`）の他の構造例を示す図である。

[図26]MMTペイロード・ヘッダ・エクステンションの他の構造例の主要な情報の内容を示す図である。

[図27]MPUペイロードの中でフラグメント化されたアクセスユニット毎に

時刻情報を送る場合について説明するための図である。

[図28]MFUの構造例 (Syntax) を示す図である。

[図29]MFUヘッダ (mfu_header()) の構造例を示す図である。

[図30]MFUヘッダ (mfu_header()) の構造例を示す図である。

[図31]MFUヘッダの構造例の主要な情報の内容を示す図である。

[図32]MFUヘッダ・エクステンション (mfu_header_extension ()) の構造例を示す図である。

[図33]MFUヘッダ・エクステンションの構造例の主要な情報の内容を示す図である。

[図34]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダ (mfu_header()) の構造例を示す図である。

[図35]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダ (mfu_header()) の構造例を示す図である。

[図36]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダの構造例の主要な情報の内容を示す図である。

[図37]MFUペイロード (mfu_payload()) の構造例およびその主要な情報の内容を示す図である。

[図38]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダ (mfu_header()) の他の構造例を示す図である。

[図39]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダ (mfu_header()) の他の構造例を示す図である。

[図40]MFUペイロードに時刻情報を挿入する場合におけるMFUヘッダの他の構造例の主要な情報の内容を示す図である。

[図41]MFUペイロード (mfu_payload()) の構造例を示す図である。

[図42]MFUペイロードの構造例の主要な情報の内容を示す図である。

[図43]タイムスタンプ・メッセージ (Timestamp Message) の構造例 (Syntax) を示す図である。

[図44]タイムスタンプ・テーブル (timestamp_table()) の構造例 (Syntax)

を示す図である。

[図45]パッケージ・アクセス・メッセージ (Package Access message) の構造例 (Syntax) を示す図である。

[図46]アクセスユニットAU (n) のデコード時刻D (n) と、表示時刻R (n) の対応関係の一例を示す図である。

[図47]時系列のオフセット値を可変長符号化するための可変長符号テーブルの一例を示す図である。

[図48]リオーダー距離Mが大きくなるほど可変長符号化による伝送容量の低減効率が高くなることを示す図である。

発明を実施するための形態

[0019] 以下、発明を実施するための形態（以下、「実施の形態」とする）について説明する。なお、説明を以下の順序で行う。

1. 実施の形態

2. 変形例

[0020] <1. 実施の形態>

[送受信システムの構成例]

図1は、実施の形態としての送受信システム10の構成例を示している。この送受信システム10は、トランスポートパケット送信装置100と、トランスポートパケット受信装置200により構成されている。

[0021] 送信装置100は、MMT構造(ISO/IEC CD 23008-1参照)のトランスポートパケット、つまりMMTパケットが含まれる伝送ストリームを生成し、この伝送ストリームをRF伝送路あるいは通信ネットワーク伝送路を通じて、受信側に送信する。この伝送ストリームには、ペイロードにビデオやオーディオの伝送メディアを含む第1のMMTパケットと、ペイロードに伝送メディアに関する情報を含む第2のMMTパケットが、少なくともフラグメントされたパケットの大きさに時分割的に多重化されている。本実施の形態において、第1のMMTパケットまたは第2のMMTパケットに、受信側でデコード時刻および表示時刻を取得するための時刻情報が挿入される。

[0022] 受信装置200は、送信側から、RF伝送路あるいは通信ネットワーク伝送路を通じて、上述の伝送ストリームを受信する。受信装置200は、伝送ストリームから取り出した伝送メディアを、時刻情報に基づいて取得されたデコード時刻および／または表示時刻を用いて処理し、画像を表示すると共に音声を出力する。

[0023] 図2は、MMTペイロード (MMT payload) の構成を概略的に示している。図2において、MMTパッケージ (MMT Package) は、MMTの論理的な概念であり、伝送素材を意味している。このMMTパッケージには、メディアであるアセット (Asset)、アセットの伝送特性 (Asset Delivery Characteristics)、パッケージに付随するメッセージ群 (Package Access)、MMTパケット・テーブルに関する情報 (MPT Packet Table)、コンポジション・インフォメーション (Composition Information) などが入っている。コンポジション・インフォメーションは、メディアの表示制御をつかさどる情報である。この例では、アセット1 (Asset1) がビデオ1 (Video1) データであり、アセット2 (Asset2) がオーディオ1 (Audio1) データであり、アセット3 (Asset3) がビデオ2 (Video2) データである。

[0024] 図2には、MMTパッケージが実際にMMTファイル (MMT File) になっている場合のファイル構成の一例を示している。このファイル構成は、基本的には、MP4のファイル構成とほぼ同等である。最初に“styp”のボックス (Box) がある。続いて、セグメントインフォメーションとしての“sidx”のボックスがある。続いて、MMT独自の“mmpu”のボックスがある。続いて、ファイル全体のメタデータとしての“moov”のボックスがある。また、続いて、“moof”および“mdat”のボックスがある。この“mdat”のボックスには、ビデオ (Video)、オーディオ (Audio)、字幕などの実際のデータが入っている。なお、“mdat”がフラグメントされる場合、フラグメントごとに“moof”のボックスがある。

[0025] MMTパッケージが伝送 (Delivery) される場合、図2に示すように、MPU (Media Processing Unit) を単位として伝送される。このMPUは、ラ

ンダムアクセスポイント（RAP：Random Access Point）から始まるものであり、一つまたは複数のアクセスユニット（AU：Access Unit）を含むものである。具体的には、例えば、一つのGOP（Group Of Picture）のピクチャが、一つのMPUの構成となることがある。このMPUは、アセット別に定義されるものとなっている。したがって、ビデオのアセットからはビデオデータのみを含むビデオのMPUが作成され、オーディオのアセットからはオーディオデータのみを含むオーディオのMPUが作成される。

[0026] 図2に示すように、MPUは、メッセージ（Message）と共に、MMTペイロードを構成するものとなる。メッセージ（Message）には、上述したコンポジション・インフォメーションなどの情報がふくまれる。MFU（MMT Fragment Unit）は、MPUが細分化、すなわちフラグメント（Fragment）化されたものである。例えば、ビデオの場合、このMFUを一つのNALユニットに相当するように設定できる。また、例えば、通信ネットワーク伝送路で送る場合、このMFUを一つまたは複数のMTUサイズ（MTU size）で構成することもできる。

[0027] 図2に示すように、MMTペイロードは、MMTパケットにより伝送される。このMMTパケットは、MMTパケットヘッダ（MMT packet header）と、MMTパケットペイロード（MMT packet payload）により構成される。そして、このMMTパケットペイロードは、MMTペイロードヘッダ（MMT payload header）と、MMTペイロードデータ（MMT payload data）により構成される。このMMTペイロードデータに、MPUやメッセージ（Message）が挿入される。

[0028] 各アセットのMPUをフラグメント化して得られたMFUは、図2に示すように、時分割的に多重される。MPUとしては、例えば15フレーム分、あるいは30フレーム分というある程度長い時間単位のものである。各MPUをフラグメント化して時分割的に多重しない場合には、ビデオデータを送っているある程度長い間、オーディオデータを送れなくなる。そのため、時間合わせのために大きなバッファ容量が必要となると共に、画像、音声の出

力までの遅延が大きくなる。各MPUをフラグメント化して時分割的に多重することで、そのような問題を解決できる。

[0029] 図3、図4、図5は、MMTファイルと、実際に伝送されるMMTパケットとの対応関係の一例を示している。MMTファイルは、上述したように、“styp”、“sidx”、“mmpu”、“moov”、“moof”、“mdat”などのボックスを有している。

[0030] MMTパケットは、MMTパケットヘッダ（MMT Hdr）と、MMTペイロード（MMT Payload）とから構成される。MMTヘッダには、パケットID（packet_id）、パケット・シーケンス・ナンバー（packet_sequence_number）、トランスミッション・タイムスタンプ（transmission_timestamp）、トランスミッション・プライオリティ（transmission_priority）、プライベート・ユーザデータ（private_user_data）などが含まれる。

[0031] パケットIDは、ビデオ、オーディオのアセット（Asset）や、コントロール（Control）系であるメッセージを識別するための識別子である。パケット・シーケンス・ナンバーは、パケット順を示す番号である。トランスミッション・タイムスタンプは、伝送のためのタイプスタンプ、すなわちMMTパケットが送信側から出ていくときの時刻である。

[0032] トランスミッション・プライオリティは、伝送路の帯域が狭くなった場合にどのMMTパケットを優先して通すかを判断するための指標となる優先度である。プライベート・ユーザデータは、ユーザが何等かの放送応用で私的に挿入し得るデータである。MMTペイロード（MMT Payload）は、MMTペイロードヘッダ（MMT PL_hdr）と、MMTペイロードデータとから構成される。MMTペイロードヘッダには、ペイロード・ヘッダ・エクステンション（payload header extension）を含めることができる。

[0033] MMTペイロードヘッダには、ペイロード・レングス（payload length）、ペイロード・タイプ（payload type）、フラグメント・タイプ（fragmentation_indicator）、フラグメント・カウント（fragment_count）、アグリゲーション・インフォ・フラグ（aggregation_info_flag）、RAPフラグ（ra

ndom_access_point_flag) などが含まれる。

[0034] また、MMTペイロードヘッダには、データ・オフセット (data_offset)、データユニット・ナンバー (numDU)、データユニット・オフセット (DU_offset)、ペイロード・シーケンス・ナンバー (payload_sequence_number)、ヘッダ・エクステンション・フィールド・フラグ (header_extension_field_flag) などが含まれる。

[0035] ペイロード・レングスは、ペイロード全体のサイズ情報である。ペイロード・タイプは、ペイロードがMPUであるか、コントロール系（メッセージ）であるかを示す情報である。一つのペイロードには、最大で64kbyteのデータを入れることができる。フラグメント・タイプは、一つのペイロードの中でMPUが完結するか否かを示す情報である。

[0036] 例えば、MPUが完結する場合は“0”が挿入され、完結しない場合、つまり所定数のMFUに細分化されている場合は、“1”、“2”、“3”のいずれかとなる。“1”はこのMMTパケットには最初のフラグメントが存在することを示す。“2”はこのMMTパケットには最初、最後のいずれでもなく、中間のフラグメントが存在することを示す。“3”はこのMMTパケットには最後のフラグメントが存在することを示す。

[0037] フラグメント・カウントは、MFUのカウント情報である。アグリゲーション・インフォ・フラグは、ペイロードに複数のMPUが入っているか否かを示すフラグ情報である。“0”は、ペイロードに一つのMPUのみが含まれることを示す。“1”は、ペイロードに複数のMPUが含まれることを示す。RAPフラグは、このMMTパケットにランダムアクセスポイント、つまりGOPの先頭ピクチャに相当するアクセスユニットが存在するか否かを示す情報である。

[0038] データ・オフセットは、ペイロード先頭位置からのペイロードデータの最初の位置までのサイズ、つまりペイロードヘッダの大きさを示す情報である。データユニット・ナンバーは、ペイロード内に存在するMPUデータユニットの個数を示す。データユニット・オフセットは、各データユニットのペ

イロードデータの先頭位置からのオフセット情報である。ペイロード・シーケンス・ナンバーは、このMMTパケットのペイロード・シーケンス・ナンバーである。ヘッダ・エクステンション・フィールド・フラグは、ペイロード・ヘッダ・エクステンションが存在するか否かを示すフラグ情報である。

[0039] 図3は、MMTパケット化の一例を示している。MMTペイロードの一つのMPUデータユニットにMMTファイルの“styp”，“sidx”，“mmpu”，“moov”，“moof”の各ボックスのメタデータが挿入されて、MMTパケットが生成される。この場合、MMTペイロードヘッダにおいて、「fragmentation_indicator」は“0”、「fragment_count」は“0”、「aggregation_info_flag」は“0”、「RAP_flag」は“1”とされる。

[0040] また、この場合、MMTペイロードの一つのMPUデータユニットにMMTファイルの“mdat”に存在する一つのMPUが挿入されて、MMTパケットが生成される。この場合、MMTペイロードヘッダにおいて、「fragmentation_indicator」は“0”、「fragment_count」は“0”、「aggregation_info_flag」は“0”、「RAP_flag」は“1”とされる。

[0041] 図4は、MMTパケット化の他の一例を示している。この場合、図3の例と同様に、MMTペイロードの一つのMPUデータユニットにMMTファイルの“styp”，“sidx”，“mmpu”，“moov”の各ボックスのメタデータが挿入されて、MMTパケットが生成される。また、この場合、MMTペイロードの複数、この例では三つのMPUデータユニットにMMTファイルの“mdat”に存在する三つのMPUが挿入されて、MMTパケットが生成される。この場合、MMTペイロードヘッダにおいて、「fragmentation_indicator」は“0”、「fragment_count」は“0”、「aggregation_info_flag」は“1”、「RAP_flag」は“1”、「numDU」は“3”とされ、「DU offset」が三つ存在する。

[0042] 図5は、MMTパケット化のさらに他の一例を示している。この場合、この場合、図3の例と同様に、MMTペイロードの一つのMPUデータユニットにMMTファイルの“styp”，“sidx”，“mmpu”，“moov”，“moof”

の各ボックスのメタデータが挿入あるいは所定の情報の変換がなされて、MMTパケットが生成される。

[0043] また、この場合、MMTペイロードの一つのMPUデータユニットにMMTファイルの“mdat”に存在する一つのアクセスユニット(AU)が、MFUヘッダ(MFU Hdr)と共に挿入されてMMTパケットが生成される。この場合、MMTペイロードヘッダにおいて、「fragmentation_indicator」は“1”、「fragment_count」はMFUのカウント値である“N”、「aggregation_info_flag」は含まれるMFUに応じて“1”～“3”のいずれか、「RAP_flag」は“1”あるいは“0”とされる。なお、MFUヘッダには、MFUのシーケンス・ナンバー(sequence_number)などの情報が含まれる。

[0044] 図6、図8は、フラグメント化されたMFUを含むMMTパケットを伝送パケットとする場合であって、そのMMTパケットがランダムアクセス(Random Access)の先頭(GOPの先頭)のパケットである場合を示している。この場合、このMMTパケットは、MMTファイルの“styp”、“sidc”、“mmpu”、“moov”、“moof”の各ボックスのメタデータが挿入あるいは所定の情報の変換がなされたMMTパケットと共に伝送される。

[0045] 例えば、“mmpu”の領域には、MFUのデータがリアルタイム系データか否かを識別する情報が含まれているが、受信側では、ランダムアクセス(Random Access)の先頭で、例えば、この識別情報によりMFUのデータがリアルタイム系データか否かの識別が可能となる。

[0046] 図7は、フラグメント化されたMFUを含むMMTパケットを伝送パケットとする場合であって、そのMMTパケットがランダムアクセス(Random Access)の非先頭のパケットである場合を示している。この場合、このMMTパケットは、上述の先頭パケットとは異なり、最小構成のMMTパケットで伝送される。

[0047] 図9は、MMTパケットの構成をツリー形式で示したものである。MMTパケットは、上述したように、MMTパケットヘッダ(MMT Packet Header)と、MMTペイロードヘッダ(MMT Payload Header)と、MMTペイロード

(MMT Payload) により構成される。MMTペイロードには、メッセージ (Message)、MPU (Media Processing Unit)、FEC修正シンボル (FEC Repair Symbol) などが含まれ、これらのシグナリングはMMTペイロードヘッダに含まれるペイロードタイプ (payload_type) により行われる。

[0048] なお、メッセージには、各種のメッセージ内容がテーブル形式で挿入される。また、MPUは、フラグメント化されて、MFU (MMT Fragment Unit) に細分化されることもある。その場合、各MFUの先頭には、MFUヘッダ (MFU Header) が付加される。MMTペイロードには、ビデオやオーディオのメディアデータに係るMPU、さらには、メタデータに係るMPUが存在する。各MPUを含むMMTパケットは、MMTパケットヘッダに存在するパケットID (Packet_ID) で識別可能とされる。

[0049] [送信装置、受信装置の概念]

図10は、トランスポートパケット送信装置100およびトランスポートパケット受信装置200の概念図を示している。送信装置100は、アセット発生部101と、エンコーダ102と、トランスポートパケット化部103と、クロック発生部104と、IP送信部105と、RF送信部106を有している。

[0050] アセット発生部101は、メディアデータとしてのビデオやオーディオのデータを発生する。このアセット発生部101は、HDD (Hard Disk Drive) や半導体メモリなどからなるデータストレージ、あるいはビデオカメラ、マイクロホンなどである。エンコーダ102は、アセット発生部101で発生されるビデオやオーディオのデータに対して符号化処理を行って送信データを生成する。

[0051] トランスポートパケット化部103は、エンコーダ102で生成される送信データをMPU単位、あるいはそれを細分化したMFU単位でペイロードに挿入した、メディアデータを含むMMTパケットを生成する他、上述したメタデータを含むMMTパケットやメッセージを含むMMTパケットを生成する。トランスポートパケット化部103は、MMTパケットを生成すると

き、受信側でデコード時刻 (Decode_Stamp) および表示時刻 (Display_Stamp) を取得するための時刻情報を、そのMMTパケットに挿入する。このMMTパケットへの時刻情報挿入の詳細については、後述する。

[0052] IP送信部105は、通信ネットワーク伝送路を使用する場合、トランスポートパケット化部103から順次出力されるMMTパケットからなる伝送ストリームをIPパケット化し、通信ネットワーク伝送路を通じて受信側に送る。RF送信部106は、RF伝送路を使用する場合、トランスポートパケット化部103から順次出力されるMMTパケットからなる伝送ストリームを適切なアダプテーションレイヤを介するなどしてRF変調し、RF伝送路を通じて受信側に送る。ここで、送信装置100は、クロック発生部104から出力されるシステム・タイム・クロックSTCに基づいて、送信データに送信タイムスタンプを付加する。

[0053] 受信装置200は、RF受信部201と、IP受信部202と、クロック再生部203と、トランスポートアンパケット化部204と、デコード・出力処理部205を有している。

[0054] RF受信部201は、送信側からRF伝送路を通じて送信されるMMTパケットの連続からなる伝送ストリームをアダプテーションレイヤの解析を経て受信し、トランスポートアンパケット化部204に出力する。また、IP受信部202は、送信側から通信ネットワーク伝送路を通じて送信されるMMTパケットの連続からなる伝送ストリームを受信し、トランスポートアンパケット化部204に出力する。

[0055] クロック発生部203は、システム・タイム・クロックSTCを発生し、トランスポートアンパケット化部204などに供給する。クロック発生部203は、送信データに付加されている送信タイムスタンプ値 (NTP値) に基づいてクロックセットを行う。あるいは、NTPパケットで供給されるタイムスタンプ値 (NTP値) に基づいてクロックセットを行う。この場合、クロック発生部203は、発生するシステム・タイム・クロックSTCとタイムスタンプ値 (NTP値) との差がある範囲に収まるように、発生するシ

ステム・タイム・クロック S T C の補正を行う。

[0056] N T P が M M T パケットヘッダの送信時タイムスタンプでなく、N T P サーバなどから取得される N T P パケットにより供給される場合、N T P のフォーマットは、図 1 1 に示すような「NTP short フォーマット」に限定されない。図示は省略するが、6 4 ビットの「NTP Timestamp Format (秒の整数部分 32 bits + 秒の小数部分 32 bits)」で供給されることもある (IETF RFC 5905 参照)。その場合、受信機内では、受信した N T P の値で受信機のデコーダクロックをセットし、メディアのタイムスタンプ値と比較される。この場合、メディアの表示あるいはデコードのタイムスタンプと受信機クロックとの間の精度の違いは、比較を行う際に受信機側で考慮される。

[0057] トランスポートアンパケット化部 2 0 4 は、R F 受信部 2 0 1 あるいは I P 受信部 2 0 2 から順次供給される M M T パケットのアンパケット化を行って、メディアデータとしての受信データを得る他、メタデータやメッセージを得る。デコード・出力処理部 2 0 5 は、トランスポートアンパケット化部 2 0 4 で得られる受信データの復号化を行ってビデオやオーディオのデータを取得し、メタデータやメッセージに基づいて、ビデオ表示やオーディオ出力を行う。

[0058] この場合、デコード・出力処理部 2 0 5 は、トランスポートアンパケット化部 2 0 4 で抽出される時刻情報に基づいて、アクセスユニット毎のデコード時刻 (Decode_Timestamp) および／または表示時刻 (Display_Timestamp) を取得し、デコードタイミングおよび表示タイミングを制御し、ビデオおよびオーディオの同期再生を実現する。このビデオおよびオーディオの同期再生の詳細については、後述する。

[0059] 図 1 0 に示す送信装置 1 0 0 および受信装置 2 0 0 の動作を簡単に説明する。最初に送信装置 1 0 0 の動作を説明する。アセット発生部 1 0 1 で発生されるビデオやオーディオのデータはエンコーダ 1 0 2 に供給される。そして、このエンコーダ 1 0 2 では、ビデオやオーディオのデータに対して符号化処理が行われて送信データ (符号化データ) が生成される。

[0060] エンコーダ 102 で生成された送信データは、トランスポートパケット化部 103 に供給される。そして、このトランスポートパケット化部 103 では、送信データを MPU 単位、あるいはそれを細分化した MFU 単位でペイロードに挿入した、メディアデータを含む MMT パケットが生成される他、上述したメタデータを含む MMT パケットやメッセージを含む MMT パケットが生成される。

[0061] トランスポートパケット化部 103 で順次生成される MMT パケットからなる伝送ストリームは、IP 送信部 105 あるいは RF 送信部 106 に供給される。IP 送信部 105 では、通信ネットワーク伝送路を使用する場合、伝送ストリームが IP パケット化され、通信ネットワーク伝送路を通じて受信側に送られる。また、RF 送信部 106 では、RF 伝送路を使用する場合、伝送ストリームが RF 変調され、RF 伝送路を通じて受信側に送られる。この際、クロック発生部 104 から出力されるシステム・タイム・クロック STC に基づいて、送信データに送信タイムスタンプが付加される。

[0062] 次に、受信装置 200 の動作を説明する。送信側から送られてくる伝送ストリームは、RF 受信部 201 あるいは IP 受信部 202 で受信され、MMT パケットが連続して含まれる伝送ストリームが取得される。この伝送ストリームは、トランスポートアンパケット化部 204 に供給される。このトランスポートアンパケット化部 204 では、MMT パケットのアンパケット化が行われ、メディアデータとしての受信データが得られる他、メタデータやメッセージ、さらには時刻情報などが得られる。これらのデータや情報は、デコード・出力処理部 205 に供給される。

[0063] デコード・出力処理部 205 では、トランスポートアンパケット化部 204 で得られる受信データの復号化が行われてビデオやオーディオのデータが取得され、メタデータやメッセージに基づいて、ビデオ表示やオーディオ出力が行われる。この場合、デコード・出力処理部 205 には、トランスポートアンパケット化部 204 で MMT パケットから抽出される時刻情報が供給される。そして、デコード・出力処理部 205 では、時刻情報に基づいて、

アクセスユニット毎のデコード時刻 (Decode_Timestamp) および／または表示時刻 (Display_Timestamp) が取得され、デコードタイミングおよび出力タイミングが制御され、ビデオおよびオーディオの同期再生が実現される。

[0064] [MMTパケットに挿入する時刻情報]

次に、MMTパケットに挿入される時刻情報について説明する。この時刻情報は、上述したように、受信側でデコード時刻 (Decode_Timestamp) および／または表示時刻 (Display_Timestamp) を取得するための情報である。MMTパケットに含まれるメディアデータは一つ以上のアクセスユニットで構成されている。MMTパケットに挿入される時刻情報は、この一つ以上のアクセスユニットにそれぞれ対応した時刻情報および／または表示時刻を取得するための情報である。

[0065] 例えば、第1の手法では、MMTパケットに挿入される時刻情報には、初期値、すなわち一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値が含まれる。また、この時刻情報には、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までのオフセット値が含まれる。また、例えば、第2の手法では、MMTパケットに挿入される時刻情報には、表示時刻、あるいは表示時刻およびデコード時刻そのものが含まれる。

[0066] 第1の手法について説明する。オフセット値は、絶対オフセット値に対応した相対オフセット値とされる。そのため、時刻情報には、相対オフセット値を絶対オフセット値に変換するための変換情報が付加される。この変換情報には、「timestamp_type」、「time tick」、「au_rate_scale」および「division_factor」などが含まれる。

[0067] 「timestamp_type」は、初期値がデコード時刻の初期値であるか表示時刻の初期値であるかを示す情報である。「time tick」は、いかなるクロックで制御するかを示す情報である。この情報により、例えば、90kHzのクロックで全てを制御することなどを示す。「au_rate_scale」は、ビデオやオーディオのアクセスユニットのレートを求めるためのスケール情報である。な

お、変換情報として、さらに、メディアデータがビデオなのかオーディオなのかを示すフラグ情報「Asset_type」も含まれる。

[0068] 例えば、ビデオの場合、上述の「time_tick」が90kHzを示すとき、「au_rate_scale」は、1500、1800、3000、3600などの値となる。それぞれ、90kHzをその値で割ることで、60Hz、50Hz、30Hz、25Hzなどのビデオレートを算出できる。

[0069] また、「division_factor」は、レートを微調整するための要素である。例えば、NTSC系のビデオにおいて、30Hzか29.94Hzかを調整するための値であり、「1」あるいは「1.001」である。

[0070] ここで、メディアデータがビデオであり、初期値がデコード時刻である場合について説明する。図12は、0番目から6番目までのアクセスユニット(AU)に対応した、デコード時刻「DTS(n)」、オフセット値「DLT(n)」、表示時刻「PTS(n)」の対応関係の一例を示している。この例では、デコード時刻の初期値からの表示時刻の初期値までのオフセット値(dlt_time)は「1」とされ、相対オフセット値とされている。

[0071] ここで、0番目のアクセスユニットはIピクチャ、1番目のアクセスユニットはPピクチャ、2番目のアクセスユニットはBピクチャ、3番目のアクセスユニットはBピクチャ、4番目のアクセスユニットはPピクチャ、5番目のアクセスユニットはBピクチャ、6番目のアクセスユニットはBピクチャである。そのため、デコード時刻に対して、表示時刻はリオーダー(Reorder)されたものとなる。

[0072] この場合、MMTパケットに挿入されて送信される時刻情報は、デコード時刻の初期値であるTS0(ここでは、TS0=0)と、相対オフセット値DLT(n)とされる。なお、図12において、「Time」はアクセスユニット毎の時刻を相対値で示している。

[0073] このような時刻情報が受信側に送信される場合、受信側では、以下のように、各アクセスユニットのデコード時刻および表示時刻を、変換情報を用いて、算出できる。この場合、初期値TS0は、最初のアクセスユニットのデ

コード時刻DTS (0) である。この初期値TS0からのオフセット時間 (dlt_time) は、以下の数式 (1) で求められる。例えば、典型例として、au_rate_scale = 1500 で、division_factor = 1.001 である場合、dlt_time = 1500 * 1.001 となる。

$$\text{dlt_time} = \text{au_rate_scale} * \text{division_factor} \cdots (1)$$

[0074] その後のアクセスユニットのデコード時刻DTS (n) は、以下の数式 (2) に示すように、オフセット時間 (dlt_time) を「time_tick」で割った値を、直前のデコード時刻DTS (n-1) に加算することで求められる。ただし、DTS (0) = TS0 である。

$$\text{DTS}(n) = \text{dlt_time} / \text{time_tick} + \text{DTS}(n-1) \cdots (2)$$

[0075] また、各アクセスユニットの表示時刻PTS (n) は、以下の数式 (3) に示すように、オフセット値DLT (n) に「dlt_time / time_tick」を掛けた値を、デコード時刻DTS (n) に加算することで求められる。

$$\text{PTS}(n) = \text{DLT}(n) * (\text{dlt_time} / \text{time_tick}) + \text{DTS}(n) \cdots (3)$$

[0076] 次に、メディアデータがビデオであり、初期値が表示時刻である場合について説明する。図13は、図12と同様の、0番目から6番目までのアクセスユニット (AU) に対応した、デコード時刻「DTS (n)」、オフセット値「DLT (n)」、表示時刻「PTS (n)」の対応関係の一例を示している。この例では、デコード時刻の初期値からの表示時刻の初期値までのオフセット値 (dlt_time) は「1」とされている。この場合も、デコード時刻に対して、表示時刻はリオーダー (Reorder) されたものとなる。

[0077] この場合、MMTパケットに挿入されて送信される時刻情報は、表示時刻の初期値であるTS0 (ここでは、TS0 = 1) と、相対オフセット値DLT (n) とされる。なお、図13において、「Time」はアクセスユニット毎の時刻を相対値で示している。

[0078] このような時刻情報が受信側に送信される場合、受信側では、以下のように、各アクセスユニットのデコード時刻および表示時刻を、変換情報を用いて、算出できる。この場合、初期値TS0は、最初のアクセスユニットの表

示時刻PTS(0)である。そのため、最初のアクセスユニットのデコード時刻DTS(0)は、以下の数式(4)に示すように、PTS(0)から、オフセット時間分、すなわちを「dlt_time」を「time_tick」で割った値にオフセット値DLT(n)を乗じた値を、減算することで求められる。

$$DTS(0) = PTS(0) - (dlt_time / time_tick) * DLT(n) \cdots (4)$$

[0079] ここで、「dlt_time」は、以下の数式(5)に示すように、「au_rate_scale」に「division_factor」を掛けることで求められる。

$$dlt_time = au_rate_scale * division_factor \cdots (5)$$

[0080] その後のアクセスユニットのデコード時刻DTS(n)は、以下の数式(6)に示すように、オフセット時間(dlt_time)を「time_tick」で割った値を、直前のデコード時刻DTS(n-1)に加算することで求められる。

$$DTS(n) = dlt_time / time_tick + DTS(n-1) \cdots (6)$$

[0081] また、各アクセスユニットの表示時刻PTS(n)は、以下の数式(7)に示すように、オフセット値DLT(n)に「dlt_time / time_tick」を掛けた値を、デコード時刻DTS(n)に加算することで求められる。

$$PTS(n) = DLT(n) * (dlt_time / time_tick) + DTS(n) \cdots (7)$$

[0082] 次に、メディアデータがオーディオであり、初期値がデコード時刻である場合について説明する。図14は、0番目から6番目までのアクセスユニット(AU)に対応した、デコード時刻「DTS(n)」、オフセット値「DLT(n)」、オーディオ出力時刻「PTS(n)」の対応関係の一例を示している。この例では、「dlt_time / time_tick」は「1」とされている。

[0083] なお、オーディオアクセスユニットは、複数のオーディオサンプルの集合体である。ここで、「dlt_time」は、以下の数式(8)に示すように、「au_rate_scale」に「division_factor」を掛けることで求められる。例えば、典型例として、オーディオサンプリング周波数が44.1kHzで、オーディオ符号化方式が1024サンプルを1つのオーディオアクセスユニットとする場合には、au_rate_scale = 2089.8で、division_factor = 1であり、dlt_time = 2089.8 * 1となる。

$$\text{dlt_time} = \text{au_rate_scale} * \text{division_factor} \cdots (8)$$

[0084] この場合、MMTパケットに挿入されて送信される時刻情報は、オーディオ出力時刻（表示時刻）の初期値であるTS0（ここでは、TS0=0）と、相対オフセット値DLT（n）とされる。なお、図14において、「Time」はアクセスユニット毎の時刻を相対値で示している。通常の場合、リオーダー（Reorder）はなく、デコードした後は、そのまま出力バッファに転送されるので、デコード時刻DTS（n）とオーディオ出力時刻PTS（n）との差分を示すDLT（n）は「0」となる。

[0085] このような時刻情報が受信側に送信される場合、受信側では、以下のように、各アクセスユニットのデコード時刻およびオーディオ出力時刻を、変換情報を用いて、算出できる。この場合、初期値TS0は、最初のアクセスユニットのオーディオ出力時刻PTS（0）であるが、そのまま、最初のアクセスユニットのデコード時刻DTS（0）ともなる。

[0086] その後のアクセスユニットのデコード時刻DTS（n）およびオーディオ出力時刻PTS（n）は、以下の数式（9）に示すように、オフセット時間（dlt_time）を「time_tick」で割った値を、直前のオーディオ出力時刻PTS（n-1）に加算することで求められる。ただし、PTS（0）=TS0である。

$$\text{PTS}(n) = \text{DTS}(n) = (\text{dlt_time} / \text{time_tick}) + \text{PTS}(n-1) \cdots (9)$$

[0087] [受信装置のデコード・出力処理部の構成例]

受信装置200のデコード・出力処理部205についてさらに説明する。図15は、デコード・出力処理部205の構成例を示している。このデコード・出力処理部205は、デマルチプレクサ301と、ビデオデコーダ302と、ビデオ表示部303と、オーディオデコーダ304と、オーディオ出力部305と、制御部306を有している。

[0088] デマルチプレクサ301は、トランスポートアンパケット化部204の出力から種々の情報あるいはデータを抽出する。すなわち、デマルチプレクサ

301は、ビデオのアクセスユニット毎の符号化ビデオデータを抽出すると共に、オーディオのアクセスユニット毎の符号化オーディオデータを抽出する。

[0089] また、デマルチプレクサ301は、ビデオ、オーディオの各アクセスユニットのデコード時刻DTS(n)および表示時刻(オーディオ出力時刻)PTS(n)を取得するための時刻情報を抽出する。この時刻情報には、上述したように初期値TS0と、相対オフセット値DLT(n)が含まれている。

[0090] また、デマルチプレクサ301は、相対オフセット値DLT(n)を絶対オフセット値に変換するための種々の情報(変換情報)を抽出する。この変換情報には、上述したように、「timestamp_type」、「time_tick」、「aurate_scale」および「division_factor」などが含まれている。

[0091] ビデオデコーダ302は、デマルチプレクサ301で抽出されるビデオの各アクセスユニットの符号化ビデオデータに対してデコード処理を行って各アクセスユニットの復号化されたビデオデータを得る。ビデオ表示部303は、ビデオデコーダ302で得られた各アクセスユニットの復号化ビデオデータに基づいてビデオ表示(画像表示)を行う。

[0092] オーディオデコーダ304は、デマルチプレクサ301で抽出されるオーディオの各アクセスユニットの符号化オーディオデータに対してデコード処理を行って各アクセスユニットの復号化されたオーディオデータを得る。オーディオ出力部305は、オーディオデコーダ304で得られた各アクセスユニットの復号化オーディオデータに基づいてオーディオ出力(音声出力)を行う。

[0093] 制御部306は、デマルチプレクサ301で抽出される時刻情報および変換情報に基づいて、ビデオ、オーディオの各アクセスユニットのデコード時刻DTS(n)および表示時刻(オーディオ出力時刻)PTS(n)を求める。そして、制御部306は、以下のように、オーディオ、ビデオの同期再生制御を行う。

[0094] すなわち、制御部306は、ビデオの各アクセスユニットのデコード時刻DTS(n)および表示時刻PTS(n)に基づいて、各アクセスユニットのデコードタイミングおよび表示タイミングを制御する。この場合、各アクセスユニットのデコードを、クロック発生部203(図7参照)で発生されるシステム・クロックSTCがデコード時刻DTS(n)となるタイミングで開始するように、ビデオデコーダ302を制御する。また、各アクセスユニットによるビデオ表示を、このシステム・クロックSTCが表示時刻PTS(n)となるタイミングで開始するように、ビデオ表示部303を制御する。

[0095] また、制御部306は、オーディオの各アクセスユニットのデコード時刻DTS(n)およびオーディオ出力時刻PTS(n)に基づいて、各アクセスユニットのデコードタイミングおよび表示タイミングを制御する。この場合、各アクセスユニットのデコードを、システム・クロックSTCがデコード時刻DTS(n)となるタイミングで開始するようにオーディオデコーダ304を制御する。また、各アクセスユニットによるオーディオ出力を、このシステム・クロックSTCがオーディオ出力時刻PTS(n)となるタイミングで開始するように、オーディオ出力部305を制御する。

[0096] 図15に示すデコード・出力制御部205の動作を簡単に説明する。トランスポートアンパケット化部204の出力は、デマルチプレクサ301に供給される。このデマルチプレクサ301では、ビデオのアクセスユニット毎の符号化ビデオデータが抽出され、ビデオデコーダ302に供給される。

[0097] ビデオデコーダ302では、ビデオの各アクセスユニットの符号化ビデオデータに対してデコード処理が行われ、各アクセスユニットの復号化されたビデオデータが得られる。このように復号化された各アクセスユニットのビデオデータは、ビデオ表示部303に供給される。ビデオ表示部303では、各アクセスユニットのビデオデータに基づいてビデオ表示(画像表示)が行われる。

[0098] また、デマルチプレクサ301では、オーディオのアクセスユニット毎の

符号化オーディオデータが抽出され、オーディオデコーダ304に供給される。オーディオデコーダ304では、オーディオの各アクセスユニットの符号化オーディオデータに対してデコード処理が行われ、各アクセスユニットの復号化されたオーディオデータが得られる。このように復号化された各アクセスユニットのオーディオデータは、オーディオ出力部305に供給される。オーディオ出力部305では、各アクセスユニットのオーディオデータに基づいてオーディオ出力（音声出力）が行われる。

[0099] また、デマルチプレクサ301では、ビデオ、オーディオの各アクセスユニットのデコード時刻 $DT S(n)$ および表示時刻（オーディオ出力時刻） $PT S(n)$ を取得するための時刻情報（初期値 $TS0$ 、相対オフセット値 $DL T(n)$ ）が抽出される。さらに、デマルチプレクサ301では、相対オフセット値 $DL T(n)$ を絶対オフセット値に変換するための種々の変換情報が抽出される。これらの時刻情報および変換情報は、制御部306に供給される。

[0100] 制御部306では、時刻情報および変換情報に基づいて、ビデオ、オーディオの各アクセスユニットのデコード時刻 $DT S(n)$ および表示時刻（オーディオ出力時刻） $PT S(n)$ が算出される。そして、制御部306では、このデコード時刻 $DT S(n)$ および表示時刻（オーディオ出力時刻） $PT S(n)$ に基づいて、オーディオ、ビデオの同期再生制御が行われる。

[0101] この場合、制御部306では、ビデオの各アクセスユニットのデコード時刻 $DT S(n)$ および表示時刻 $PT S(n)$ に基づいて、ビデオの各アクセスユニットのデコードタイミングおよび表示タイミングが制御される。また、制御部306では、オーディオの各アクセスユニットのデコード時刻 $DT S(n)$ およびオーディオ出力時刻 $PT S(n)$ に基づいて、オーディオの各アクセスユニットのデコードタイミングおよび出力タイミングが制御される。

[0102] なお、上述の説明では、MMTパケットに挿入される時刻情報が第1の手法による場合、つまり時刻情報が、デコード時刻あるいは表示時刻の初期値

と、各アクセスユニットに対応したオフセット値である場合を説明した。詳細説明は省略するが、MMT packetsに挿入される時刻情報が第2の手法による場合、つまり時刻情報が、各アクセスユニットの表示時刻、あるいは表示時刻およびデコード時刻そのものである場合も考えられる。その場合、制御部306では、それらの時刻がそのまま用いられる。

[0103] 図16は、制御部306におけるAV同期再生制御の一例を示すタイミングチャートである。図16(a)は、クロック発生部203で発生されるシステム・クロックSTCを示している。このシステム・クロックSTCは、上述したように、送信側から送られてくる送信タイムスタンプ(NTP値)あるいは、NTP packetsで供給されるタイムスタンプ値(NTP値)によりクロックセットが行われる。

[0104] 図16(b)に示すように、システム・クロックSTCがビデオの各アクセスユニットのデコード時刻DTS(n)となる毎に、ビデオの各アクセスユニットのデコードが開始される。そして、図16(c)に示すように、システム・タイム・クロックSTCがビデオの各アクセスユニットの表示時刻PTS(n)となる毎に、ビデオの各アクセスユニットのビデオ表示(画像表示)が開始される。

[0105] 図16(d)に示すように、システム・クロックSTCがオーディオの各アクセスユニットのデコード時刻DTS(n)となる毎に、オーディオの各アクセスユニットのデコードが開始されると共に、オーディオ出力(音声出力)が開始される。図16(e)は、オーディオサンプル出力を示している。

[0106] なお、MFUにフラグメント化されている packets が、伝送上でドロップされた場合、あるいは、受信装置200内で表示に使わないと判断された場合、受信側で再生されるタイムスタンプ・テーブルのアクセスユニットのカウンタと、実際に受信するアクセスユニットと関係は、図17に示すようになる。MFUヘッダのシーケンス・ナンバー(sequence_number)によって整順されたMFUデータとして受信するアクセスユニット(AU)をタイムスタンプ・テーブル上で検査し、圧縮バッファからはDTS(n)のタイミング

でデコーダに転送する。また、デコードバッファからPTS (n) のタイミングで表示用プロセスへ転送する。このようにして、受信しないアクセスユニットあるいは、表示させないと判断したアクセスユニットに対しての、タイムスタンプ (Timestamp) を参照するか、しないかを判断する。

[0107] [ファイルから伝送パケットに変換する際の時刻情報の生成方法]

ファイルの状態から伝送パケットへ変換する際、MMTファイルから以下のように取得された情報に基づいて、MMTパケットに挿入すべき時刻情報 (受信側でデコード時刻および／または表示時刻を取得するための時刻情報) が生成される。ファイル内の情報から、MMTパケットに挿入する時刻情報の生成は、トランスポートパケット化部103で行われる。

[0108] 最初に、ファイル全体に対して「moov」情報を用いる場合について説明する。この場合、“Moov”ボックスの配下の‘stts’, ‘ctts’のボックスで、ファイル内に含まれる全サンプルのデコード時刻、表示時刻が供給される。この際、‘stts’ (decoding time to sample)のボックスにより、デコード時刻が与えられる。また、‘ctts’ (composition time to sample)のボックスによりデコード時刻と表示時刻の差分が与えられる。また、‘stss’ (sync sample table)のボックスでは、ランダムアクセスのサンプル位置が示される。

[0109] 次に、フラグメントごとに「moof」情報を用いる場合について説明する。この場合、“Moof”ボックスの配下の、‘trun’, ‘tfra’のボックスでフラグメントごとのサンプルのデコード時刻、表示時刻が供給される。この際、‘trun’ (track fragment run)のボックスにより、フラグメントの先頭位置からのオフセット位置と、サンプルのデコード時刻、表示時刻とのオフセット (ビデオの場合はフレーム数、オーディオの場合はオーディオサンプル数) が供給される。また、‘tfra’ (track fragment random access)のボックスにより、ランダムアクセス位置と、そのサンプルのデコード時刻が供給され、また、サンプルごとのデコード時刻と表示時刻との差分値から、表示時刻も分かる。

[0110] [時刻情報の挿入位置]

次に、時刻情報の挿入方法について説明する。図18は、MMTパケットに時刻情報を挿入する方法を概略的に示している。図18(a)は、MPUペイロードヘッダ(MPU payload header)の中で時刻情報を送る場合を示している。この場合、MPUのペイロードが複数のアクセスユニット(AU)を含むときは、ペイロード・ヘッダ・エクステンション(payload header extension)に、その複数のアクセスユニット分の時刻情報を挿入する。一方、MPUペイロードに一つのアクセスユニットを持つような構成では、ペイロード・ヘッダ・エクステンションに、個々のアクセスユニットに対して時刻情報を挿入する。

[0111] 図18(b)は、MPUペイロードの中でフラグメント化されたアクセスユニット(AU)毎に時刻情報を送る場合を示している。この場合、MFUに一つのアクセスユニットを持つような構成として、MFU・ヘッダ・エクステンション(MFU header extension)に、個々のアクセスユニットに対して時刻情報を挿入する。あるいは、MFUペイロード(MFU payload)に、個々のアクセスユニットに対して時刻情報を挿入する。また、図18(c)は、メッセージ(Message)で時刻情報を送る場合を示している。この場合、少なくとも、RAP(ランダムアクセスポイント)単位で、対象のパケットID(packet_id)で関連付けするメディアに対しての時刻情報を持つメッセージを伝送する。

[0112] 図19は、送受信遅延を低くするために、各アクセスユニットに対応して時刻情報(受信側でデコード時刻および／または表示時刻を取得するための時刻情報)を挿入する場合のパケット構成の一例を示している。図19(a)は、MMTペイロードヘッダに時刻情報を挿入する場合のパケット構成例である。図19(b)は、MFU内に時刻情報を挿入する場合のパケット構成例である。さらに、図19(c)は、MMTメッセージに時刻情報を挿入する場合のパケット構成例である。

[0113] なお、各パケット構成例において、対象のアクセスユニットを持つMMT

パケットがランダムアクセスの先頭パケット、つまりランダムアクセスポイントのパケットである場合には、MMTファイルの“styp”、“sidx”、“mmpu”、“moov”の各ボックスのメタデータが挿入されたMMTパケットと共に伝送される。一方、このMMTパケットがランダムアクセスの非先頭パケット、つまり非ランダムアクセスポイントのパケットである場合には、上述のメタデータが挿入されたMMTパケットは伝送されない

[0114] MPUペイロードヘッダ (MPU payload header) の中で時刻情報を送る場合についてさらに説明する。図20は、MMTパケットの全体の構造例 (Syntax) を示している。MMTパケット (mmt_packet()) には、MMTパケットヘッダ (mmt_header()) と、MMTペイロード (mmt_payload()) が含まれている。さらに、MMTペイロードには、MMTペイロードヘッダ (mmt_payload_header()) と、MMTペイロードデータ (mmt_payload_data()) が含まれている。

[0115] 図21は、MMTパケットヘッダ (mmt_header()) の構造例 (Syntax) を示している。詳細説明は省略するが、このMMTパケットヘッダには、上述したように、パケットID (packet_id)、パケット・シーケンス・ナンバー (packet_sequence_number)、トランスミッション・タイムスタンプ (transmission_timestamp)、トランスミッション・プライオリティ (transmission Priority)、プライベート・ユーザデータ (private user_data) などが含まれている。

[0116] 図22は、MMTペイロードヘッダ (mmt_payload_header()) の構造例 (Syntax) を示している。詳細説明は省略するが、このMMTペイロードヘッダには、上述したように、ペイロード・レングス (payload_length)、ペイロード・タイプ (payload_type)、フラグメント・タイプ (fragment_type)、フラグメント・カウント (fragment_count)、アグリゲーション・インフォ・フラグ (aggregation_info_flag)、RAPフラグ (random_access_point_flag)、データ・オフセット (data_offset)、データユニット・ナンバー (numDU)、データユニット・オフセット (DU_offset)、ペイロード・シー

ケンス・ナンバー (payload_seq_number)、ヘッダ・エクステンション・フィールド・フラグ (header_extension_field_flag) などが含まれている。

[0117] そして、ヘッダ・エクステンション・フィールド・フラグが“1”であるとき、このMMTペイロードヘッダには、MMTペイロード・ヘッダ・エクステンション (mmt_payload_header_extension()) がさらに含まれる。

[0118] 図23は、MMTペイロード・ヘッダ・エクステンション (mmt_payload_header_extension()) の構造例 (Syntax) を示している。この構造例は、MPUペイロードヘッダ (MPU payload header) の中で時刻情報を送る場合に対応している。図24は、この構造例の主要な情報の内容 (Semantics) を示している。

[0119] 「payload_header_extension_type」の16ビットフィールドは、MMTペイロード・ヘッダ・エクステンションのタイプを示す。例えば、“0x0001”は、処理のための時刻情報 (タイプスタンプ) を提供することを示す。「payload_header_extension_length」の16ビットフィールドは、MMTペイロード・ヘッダ・エクステンションのサイズを示す。「asset_type」2ビットフィールドは、アセットタイプを示す。例えば、“01”はビデオを示し、“10”はオーディオを示す。

[0120] 「time_tick」の2ビットフィールドは、いかなるクロックで制御するかを示す。“01”は90kHzの精度のクロックの値を示す。“10”は「NTP short time」であることを示す。「au_rate_scale」の3ビットフィールドは、ビデオやオーディオのアクセスユニットのレートを求めるためのスケール情報である。

[0121] アセットタイプがビデオの場合、“000”は3750の値を示し、この値によりレートとして24Hzを求めることができる。また、“001”は3600の値を示し、この値によりレートとして25Hzを求めることができる。また、“010”は3000の値を示し、この値によりレートとして30Hzを求めることができる。

[0122] また、“011”は1800の値を示し、この値によりレートとして50

H zを求めることができる。また、“1 0 0”は1 5 0 0の値を示し、この値によりレートとして6 0 H zを求めることができる。また、“1 0 1”は9 0 0の値を示し、この値によりレートとして1 0 0 H zを求めることができる。また、“1 1 0”は7 5 0の値を示し、この値によりレートとして1 2 0 H zを求めることができる。なお、“1 1 1”はリザーブとされる。

[0123] 一方、アセットタイプがオーディオの場合、“0 0 0”は1 9 2 0の値を示し、この値によりレートとして4 8 K H z * 1 0 2 4を求めることができる。また、“0 0 1”は2 0 8 9 . 8の値を示し、この値によりレートとして4 4 . 1 K H z * 1 0 2 4を求めることができる。また、“0 1 0”は2 8 8 0の値を示し、この値によりレートとして3 2 K H z * 1 0 2 4を求めることができる。なお、その他はリザーブとされる。

[0124] 「division_factor」の2ビットフィールドは、レートを微調整するための要素である。アセットタイプがビデオの場合、“0 0”は1を示し、“0 1”は1 . 0 0 1を示す。また、アセットタイプがオーディオの場合、“0 0”は1を示し、“0 1”は2を示す。

[0125] 「timestamp_type」の1ビットフィールドは、初期値がデコード時刻の初期値であるか表示時刻の初期値であるかを示す情報である。“1”はデコード時刻 (decode_timestamp) であることを示し、“0”は表示時刻 (display_timestamp) であることを示す。

[0126] 「timestamp_for_processing」の3 2ビットフィールドは、初期値を示す。この場合、「time_tick」が“0 1”であるときは、9 0 K H z精度のものとなる。また、「time_tick」が“1 0”であるときは、R F C 5 0 5 9で規定される「N T P short timestamp」となる。

[0127] 「au_count_in_mpu」の1 0ビットフィールドは、M P Uに含まれるアクセスユニット (AU) の個数を示す。「delta_sequence_type」の1ビットフィールドは、オフセット値が可変長符号であるか、8ビット固定長であることを示す。“1”は可変長符号であることを示し、“0”は8ビット固定長であることを示す。なお、オフセット値の可変長符号化については後述する。

- [0128] 「delta_fixed_length_code」の8ビットフィールドは、8ビット固定長のオフセット値を挿入する領域である。「delta_variable_length_code」の可変長領域は、可変長符号としてのオフセット値を挿入する領域である。なお、「trailing_filler()」は、MMTペイロード・ヘッダ・エクステンション (mmt_payload_header_extension()) のサイズがバイトアラインされるように、「delta_variable_length_code」の累積に対し、適宜、1～7ビットの連続する“0”を挿入するものである。
- [0129] なお、図24の内容 (Semantics) に追加することで、ビデオの場合、time_tick=60Hz、au_rate_scale=1を割り当て、オーディオの場合、time_tick=44,1KHz、au_rate_scale=1を割り当てることもできる。
- [0130] なお、上述では、MMTペイロード・ヘッダ・エクステンションに、デコード時刻あるいは表示時刻と、オフセット値を挿入する、第1の手法に対応した構造例を示した。このMMTペイロード・ヘッダ・エクステンションに表示時刻、あるいは表示時刻およびデコード時刻を挿入する、第2の手法に対応した構造例も考えられる。
- [0131] 図25は、その場合におけるMMTペイロード・ヘッダ・エクステンション (mmt_payload_header_extension()) の構造例 (Syntax) を示している。この構造例も、MPUペイロードヘッダ (MPU payload header) の中で時刻情報を送る場合に対応している。図26は、この構造例の主要な情報の内容 (Semantics) を示している。
- [0132] 「payload_header_extension_type」の16ビットフィールドは、MMTペイロード・ヘッダ・エクステンションのタイプを示す。例えば、“0x01”は、NTP short time フォーマットの表示タイムスタンプ (表示時刻) を供給することを示す。“0x02”は、NTP short time フォーマットの表示タイムスタンプと、デコードタイムスタンプ (デコード時刻) とを供給することを示す。“0x03”は、90KHz精度での表示タイムスタンプを供給することを示す。“0x04”は、90KHz精度での表示タイムスタンプと、デコードタイムスタンプとを供給することを示す。

- [0133] 「payload_header_extension_length」の16ビットフィールドは、MMTペイロード・ヘッダ・エクステンションのサイズを示す。「presentation_timestamp」の32ビットフィールドは、表示タイムスタンプ（表示時刻）の値を示す。「decoding_timestamp」の32ビットフィールドは、デコードタイムスタンプ（デコード時刻）の値を示す。
- [0134] 次に、MPUペイロードの中でフラグメント化されたアクセスユニット（AU）毎に時刻情報を送る場合についてさらに説明する。この場合、例えば、図27に示すように、アクセスユニット（AU）毎に、表示時刻（表示タイムスタンプ）PTSと、オフセット値DLTを、MFUヘッダ・エクステンション（mfu_header_extension）の領域に挿入して伝送する。なお、図示しないが、表示時刻（表示タイムスタンプ）PTSの代わりに、デコード時刻（デコードタイムスタンプ）DTSを伝送することも可能である。
- [0135] 図28は、MFUの構造例（Syntax）を示している。MFUには、MFUヘッダ（mfu_header()）と、MFUメディアデータ（mfu_media_data()）が含まれている。図29、図30は、MFUヘッダ（mfu_header()）の構造例（Syntax）を示している。図31は、この構造例の主要な情報の内容（Semantics）を示している。
- [0136] 「sequence_number」の32ビットフィールドは、MPU内のMFUのシーケンス番号である。「trackref_index」の8ビットフィールドは、メディアトラックの番号である。「sample_number」の32ビットフィールドは、当MFUが属するサンプル（アクセスユニット）のmoofボックス内の順番を示す。「priority」の8ビットフィールドは、MPU内の当MFUの優先度を示す。「dependency_counter」の8ビットフィールドは、当MFUにデコードが依存しているMFUの数を示す。
- [0137] 「offset」の16ビットフィールドは、mdatボックスからのオフセットを示す。「length」の32ビットフィールドは、MFUの大きさを示す。「multiLayer_flag」の1ビットフィールドは、マルチレイヤ情報の有無を示す。
- 「mfu_header_extension_flag」の1ビットフィールドは、MFUエクステン

ション (mfu_extension) の有無を示す。

- [0138] 「dependency_id」、「depth_flag」、「temporal_id」、「quality_id」、「priority_id」、「view_id」、「layer_id」は、MFU同士の種々の依存関係を示すID群である。「Item_ID」は、ファイルのIDを示す。MFUヘッダ・エクステンション・フラグ (mfu_header_extension_flag) が“1”であるとき、MFUヘッダ・エクステンション (mfu_header_extension()) が存在する。
- [0139] 図32は、MFUヘッダ・エクステンション (mfu_header_extension()) の構造例 (Syntax) を示している。図33は、この構造例の主要な情報の内容 (Semantics) を示している。「asset_type」2ビットフィールドは、アセットタイプを示す。例えば、“01”はビデオを示し、“10”はオーディオを示す。
- [0140] 「time_tick」の2ビットフィールドは、いかなるクロックで制御するかを示す。“01”は90kHzの精度のクロックで全てを制御することを示す。“10”は「NTP short time」であることを示す。「au_rate_scale」の10ビットフィールドは、ビデオやオーディオのアクセスユニットのレートを求めるためのスケール情報である。
- [0141] アセットタイプがビデオの場合、“000”は3750の値を示し、この値によりレートとして24Hzを求めることができる。また、“001”は3600の値を示し、この値によりレートとして25Hzを求めることができる。また、“010”は3000の値を示し、この値によりレートとして30Hzを求めることができる。
- [0142] また、“011”は1800の値を示し、この値によりレートとして50Hzを求めることができる。また、“100”は1500の値を示し、この値によりレートとして60Hzを求めることができる。また、“101”は900の値を示し、この値によりレートとして100Hzを求めることができる。また、“110”は750の値を示し、この値によりレートとして120Hzを求めることができる。なお、“111”はリザーブとされる。

- [0143] 一方、アセットタイプがオーディオの場合、“000”は1920の値を示し、この値によりレートとして $48\text{ KHz} * 1024$ を求めることができる。また、“001”は2089.8の値を示し、この値によりレートとして $44.1\text{ KHz} * 1024$ を求めることができる。また、“010”は2880の値を示し、この値によりレートとして $32\text{ KHz} * 1024$ を求めることができる。なお、その他はリザーブとされる。
- [0144] 「division_factor」の2ビットフィールドは、レートを微調整するための要素である。アセットタイプがビデオの場合、“00”は1を示し、“01”は1.001を示す。また、アセットタイプがオーディオの場合、“00”は1を示し、“01”は2を示す。
- [0145] 「timestamp_for_processing」の32ビットフィールドは、対応するアクセスユニットの時刻を示す。この場合、「time_tick」が“01”であるときは、90KHz精度のものとなる。また、「time_tick」が“10”であるときは、RFC 5059で規定される「NTP short timestamp」となる。
- [0146] 「timestamp_type」の1ビットフィールドは、上述の「timestamp_for_processing」で示される時刻がデコード時刻であるか表示時刻であることを示す。“1”はデコード時刻 (decode_timestamp) であることを示し、“0”は表示時刻 (display_timestamp) であることを示す。「delta_fixed_length_code」の8ビットフィールドは、8ビット固定長のオフセット値を挿入する領域である。
- [0147] なお、上述では、MFUヘッダ・エクステンションに、時刻情報（例えば、表示時刻PTSあるいはデコード時刻DTSと、オフセット値DLT）を挿入する構造例を示した。MFUペイロードに時刻情報を挿入する構造例も考えられる。図34、図35は、その場合におけるMFUヘッダ (mfu_header()) の構造例 (Syntax) を示している。図36は、この構造例の主要な情報の内容 (Semantics) を示している。
- [0148] 「timestamp_type」の2ビットフィールドは、MPUペイロードの先頭にタイムスタンプが供給されるかどうかを示す。“00”は、タイムスタンプ

は供給されないことを示す。この場合、MPUペイロードの最初のバイト（1st バイト）がフラグメントされたMPUペイロードの先頭バイトと一致する。“01”は、MPUペイロードの先頭に表示タイムスタンプが供給されることを示す。この場合、表示タイムスタンプの直後にフラグメントされたMPUペイロードの先頭バイトが供給される。“10”は、MFUペイロードの先頭にデコードタイムスタンプと表示タイムスタンプの双方が供給されることを示す。この場合、タイムスタンプの直後にフラグメントされたMPUペイロードの先頭バイトが供給される。

[0149] 「time_tick」の1ビットフィールドは、タイムスタンプの値の精度を示す。“0”は、90000Hz精度であることを示す。“1”は、NTP short time フォーマットであることを示す。

[0150] 図37は、MFUペイロード(mfu_payload())の構造例(Syntax)およびその主要な情報の内容(Semantics)を示している。「presentation_timestamp」の32ビットフィールドは、表示タイムスタンプ(表示時刻)の値を示す。「decoding_timestamp」の32ビットフィールドは、デコードタイムスタンプ(デコード時刻)の値を示す。MFUヘッダにおける「timestamp_type」が“01”であるとき、ペイロードデータ(payload_data)の直前に「presentation_timestamp」の32ビットフィールドが存在する。また、MFUヘッダにおける「timestamp_type」が“10”であるとき、ペイロードデータ(payload_data)の直前に「presentation_timestamp」の32ビットフィールドおよび「decoding_timestamp」の32ビットフィールドが存在する。

[0151] 図38、図39は、MFUペイロードに時刻情報を挿入する場合における、MFUヘッダ(mfu_header())の他の構造例(Syntax)を示している。図40は、この構造例の主要な情報の内容(Semantics)を示している。「payload_header_flag」の1ビットフラグ情報は、MFUペイロードの先頭に、MFUペイロードヘッダが存在するかどうかを示す。“0”は、MFUペイロードヘッダは存在しないことを示す。この場合、MFUペイロードの最初のバイト(1st バイト)はフラグメントされたMPUペイロードの先頭バイト

と一致する。“1”は、MFUペイロードヘッダが存在することを示す。この場合、MFUペイロードヘッダの直後にフラグメントされたMPUペイロードの先頭バイトが供給される。

[0152] 図41は、MFUペイロード(mfu_payload())の構造例(Syntax)を示している。図42は、この構造例の主要な情報の内容(Semantics)を示している。「mfu_payload_header_type」の8ビットフィールドは、MFUペイロードヘッダのタイプを示す。例えば、“0x01”は、NTP short time フォーマットの表示タイムスタンプ(表示時刻)を供給することを示す。“0x02”は、NTP short time フォーマットの表示タイムスタンプと、デコードタイムスタンプ(デコード時刻)とを供給することを示す。“0x03”は、90KHz精度での表示タイムスタンプを供給することを示す。“0x04”は、90KHz精度での表示タイムスタンプと、デコードタイムスタンプとを供給することを示す。

[0153] 「mfu_payload_header_length」の8ビットフィールドは、本エレメント以後のMFUペイロードヘッダのサイズをバイト数で示す。「presentation_timestamp」の32ビットフィールドは、表示タイムスタンプ(表示時刻)の値を示す。「decoding_timestamp」の32ビットフィールドは、デコードタイムスタンプ(デコード時刻)の値を示す。

[0154] 次に、メッセージ(Message)で時刻情報を送る場合についてさらに説明する。図43は、タイムスタンプ・メッセージ(Timestamp Message)の構造例(Syntax)を示している。このタイムスタンプ・メッセージは、MPUに含まれる各アクセスユニット(AU)のデコード時刻(decode_timestamp)、表示時刻(display_timestamp)を受信側で取得するために、受信側の時刻情報を供給するためのメッセージである。

[0155] このタイムスタンプ・メッセージには、メッセージ内容が配置されるタイムスタンプ・テーブル(timestamp_table())が含まれる。図44は、このタイムスタンプ・テーブル(timestamp_table())の構造例(Syntax)を示している。詳細説明は省略するが、このタイムスタンプ・テーブルには、上述し

たMMTペイロード・ヘッダ・エクステンション (mmt_payload_header_extension()) (図23、図24参照)と同様の情報(時刻情報、変換情報)が含まれる。

[0156] このタイムスタンプ・メッセージには、パケットID (packet_ID)が含まれ、アセット (asset) レベルでの関連付が行われる。その他、このタイムスタンプ・メッセージには、従来のテーブル (table) 構成に従い、テーブルID (table_id)、バージョン (version)、レングス (length) などの情報が追加されている。ここで、テーブルIDやバージョンは、システムでユニークに割り当てられるものである。また、レングスは、当テーブル全体のサイズを示す。

[0157] このタイムスタンプ・テーブル (timestamp_table) は、メッセージ (message) 群を束ねるメッセージである。パッケージ・アクセス・メッセージ (Package Access message) の中のメッセージペイロード (message payload ()) で伝送することも可能である。図45は、パッケージ・アクセス・メッセージ (Package Access message) の構造例 (Syntax) を示している。

[0158] その場合、パッケージ・アクセス・メッセージとしては、図示のように、ヘッダ部分にパッケージ・アクセス・メッセージを示す「message_id」から始まり、「version」、「length」が続く。そして、「number_of_tables」でテーブル (table) の数が示された後、そのテーブルの数だけループ (loop) があり、各ループの中は、「table_id」から始まり、「table_version」、「table_length」が続く。「table_id」は、タイムスタンプ・テーブル (timestamp_table) の中の「table_id」と一致する。「table_length」は、「timestamp_table」を含み、他の「table」を含んだサイズを示す。その後、「message_payload」の中に、「timestamp_table」がそのまま供給される。

[0159] [オフセット値の可変長符号化]

次に、オフセット値DLT (n) の可変長符号化について説明する。上述したように、アクセスユニット (AU) 毎のデコード時刻および表示時刻を受信側で再現するために、各アクセスユニットのオフセット値を時系列的に伝

送することが行われる。このオフセット値は、配信順（＝デコード順）の系列を表示順に整列する際に有効となるものであり、主にビデオに関するピクチャ・リオーダー（picture reorder）を正しく行い、かつ同期表示を確保する目的で参照される。

[0160] 図46は、アクセスユニットAU（ n ）のデコード時刻D（ n ）と、表示時刻R（ n ）の対応関係の一例を示している。この例の場合、リオーダー距離 $M=3$ であり、以下の値をアクセスユニットの数だけ符号化することになる。

D（0）→R（0）のオフセット値DLT＝＋1

D（1）→R（1）のオフセット値DLT＝＋3

D（2）→R（2）のオフセット値DLT＝0

D（3）→R（3）のオフセット値DLT＝0

[0161] よって、オフセット値DLTの時系列表現は、＋1，＋3，0，0，＋3，0，0，3，・・・となる。これは、リオーダー距離 M に依存するものであり、リオーダー距離 M が大きくなれば、“0”を入れる割合が増大する。本実施の形態では、このことを利用して、0発生確率に応じて有効な符号サイズを確保して、伝送容量に合わせる目的で可変長符号化を行う。

[0162] 図47は、時系列のオフセット値を可変長符号化するための可変長符号テーブルの一例を示している。高確率で発生する“0”に対してショートワード（short word）が割り当てられ、リオーダー距離 M の単位で発生する値には、ロングワード（long word）が割り当てられる。このような可変長符号化により、時系列のオフセット値の伝送容量を効率よく低減できる。

[0163] なお、リオーダー距離 M が大きくなるほどBピクチャの割合が大きくなるので、このような可変長符号化による伝送容量の低減効率は大きくなっていく。図48は、このことを示している。例えば、リオーダー距離 M が3の場合、「IPBB」の4個のアクセスユニットのオフセット値の伝送容量は10ビットとなり、平均すると、10／4ビットとなる。これに対して、例えば、リオーダー距離 M が6の場合、「IPBBBBBB」の7個のアクセスユ

ニットのオフセット値の伝送容量は13ビットとなり、平均すると、13／7ビットとなる。

[0164] 上述したように、図1に示す送受信システム10においては、送信装置100から受信装置200に送られるMMTパケットに、受信側で各アクセスユニットのデコード時刻および/または表示時刻を取得するための時刻情報が挿入されるものである。したがって、例えば、受信側でデコード時刻および/または表示時刻を取得するための時刻情報の伝送を良好に行うことができる。

[0165] また、図1に示す送受信システム10においては、時刻情報は、例えば、一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値（初期値）と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までのオフセット値とされるものである。また、このオフセット値は絶対オフセット値ではなく、相対オフセット値とされる。したがって、時刻情報の伝送を効率よく行うことができる。

[0166] また、図1に示す送受信システム10においては、時系列のオフセットを可変長符号化して伝送するものである。したがって、時刻情報の伝送容量の低減を図ることができる。

[0167] <2. 変形例>

なお、上述実施の形態においては、ファイル内の時刻情報をMMTパケットに反映する方法を記述したが、“moov”内、あるいは“moof”内の時刻情報をそのままMMTペイロードヘッダあるいはMFU内の領域、またはメッセージにて供給することとしても類似の効果が得られる。

[0168] また、上述実施の形態においては、本技術を、MMT構造のトランスポートパケット、つまりMMTパケットが含まれる伝送ストリームをRF伝送路あるいは通信ネットワーク伝送路を通じて伝送する送受信システム10に適用したものである。本技術は、同様の伝送ストリームをRF伝送路あるいは通信ネットワーク伝送路を通じて伝送する送受信システムにも同様に適用できることは勿論である。

[0169] また、本技術は、以下のような構成を取ることにもできる。

(1) ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを生成する伝送ストリーム生成部と、

上記伝送ストリームを所定の伝送路を通じて受信側に送信する伝送ストリーム送信部と、

上記第1の伝送パケットまたは上記第2の伝送パケットに、上記受信側でデコード時刻および/または表示時刻を取得するための時刻情報を挿入する時刻情報挿入部とを備える

送信装置。

(2) 上記第1の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、

上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットにそれぞれ対応したデコード時刻および/または表示時刻を取得するための情報である

前記(1)に記載の送信装置。

(3) 上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までのオフセット値とを有する

前記(2)に記載の送信装置。

(4) 上記オフセット値は、絶対オフセット値に対応した相対オフセット値であり、

上記時刻情報挿入部で挿入される時刻情報には、上記相対オフセット値を上記絶対オフセット値に変換するための変換情報が付加される

前記(3)に記載の送信装置。

(5) 上記時刻情報挿入部は、上記相対オフセット値を可変長符号化した後に挿入する

前記（４）に記載の送信装置。

（６）上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットのそれぞれに対応した表示時刻、あるいは表示時刻およびデコード時刻である

前記（２）に記載の送信装置。

（７）上記第１の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記時刻情報挿入部は、上記ペイロードヘッダ部に、上記時刻情報を挿入する

前記（１）から（６）のいずれかに記載の送信装置。

（８）上記第１の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記ペイロード本体部には、上記伝送メディアがフラグメント化された一つ以上のアクセスユニットをそれぞれ含むフラグメントペイロードがフラグメントヘッダを伴って配置され、

上記時刻情報挿入部は、上記フラグメントヘッダに、対応するアクセスユニットの時刻情報を挿入する

前記（１）から（６）のいずれかに記載の送信装置。

（９）上記第１の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記ペイロード本体部には、上記伝送メディアがフラグメント化された一つ以上のアクセスユニットをそれぞれ含むフラグメントペイロードがフラグメントヘッダを伴って配置され、

上記時刻情報挿入部は、上記フラグメントペイロードに、対応するアクセスユニットの時刻情報を挿入する

前記（１）から（６）のいずれかに記載の送信装置。

（１０）上記第２の伝送パケットのペイロード部は、ペイロードヘッダ部とペイロード本体部とからなり、

上記時刻情報挿入部は、上記ペイロード本体部に、上記時刻情報を挿入する

前記（１）から（６）のいずれかに記載の送信装置。

（１１）上記所定の伝送路は、ＲＦ伝送路または通信ネットワーク伝送路である

前記（１）から（１０）のいずれかに記載の送信装置。

（１２）上記第１の伝送パケットおよび上記第２の伝送パケットは、ＭＭＴパケットである

前記（１）から（１１）のいずれかに記載の送信装置。

（１３）ペイロードに伝送メディアを含む第１の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第２の伝送パケットとが時分割的に多重化された伝送ストリームを生成する伝送ストリーム生成ステップと、

上記伝送ストリームを所定の伝送路を通じて受信側に送信する伝送ストリーム送信ステップと、

上記第１の伝送パケットまたは上記第２の伝送パケットに、上記受信側でデコード時刻および/または表示時刻を取得するための時刻情報を挿入する時刻情報挿入ステップとを備える

送信方法。

（１４）ペイロードに伝送メディアを含む第１の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第２の伝送パケットとが時分割的に多重化された伝送ストリームを、送信側から所定の伝送路を通じて受信する伝送ストリーム受信部を備え、

上記第１の伝送パケットまたは上記第２の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、

上記伝送ストリームから取り出された上記伝送メディアを、上記時刻情報に基づいて取得された上記デコード時刻および/または上記表示時刻を用いて処理する伝送メディア処理部をさらに備える

受信装置。

(15) 上記第1の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、

上記時刻情報は、上記一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までの絶対オフセット値に対応した相対オフセット値であり、

上記相対オフセット値を上記絶対オフセット値に変換するオフセット値変換部をさらに備える

前記(14)に記載の受信装置。

(16) ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを、送信側から所定の伝送路を通じて受信する伝送ストリーム受信ステップを備え、

上記第1の伝送パケットまたは上記第2の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、

上記伝送ストリームから取り出された上記伝送メディアを、上記時刻情報に基づいて取得された上記デコード時刻および/または表示時刻を用いて処理する伝送メディア処理ステップをさらに備える

受信方法。

[0170] 本技術の主な特徴は、MMTパケットに、複数のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までの相対オフセット値とを挿入することで、時刻情報の伝送を効率よく行い得るようにしたことである(図15、図18参照)。

符号の説明

[0171] 10・・・送受信システム

100・・・トランスポートパケット送信装置

- 1 0 1 . . . アセット発生部
- 1 0 2 . . . エンコーダ
- 1 0 3 . . . トランスポートパケット化部
- 1 0 4 . . . クロック発生部
- 1 0 5 . . . I P 送信部
- 1 0 6 . . . R F 送信部
- 2 0 0 . . . トランスポートパケット受信装置
- 2 0 1 . . . R F 受信部
- 2 0 2 . . . I P 受信部
- 2 0 3 . . . クロック発生部
- 2 0 4 . . . トランスポートアンパケット化部
- 2 0 5 . . . デコード・出力処理部
- 3 0 1 . . . デマルチプレクサ
- 3 0 2 . . . ビデオデコーダ
- 3 0 3 . . . ビデオ表示部
- 3 0 4 . . . オーディオデコーダ
- 3 0 5 . . . オーディオ出力部
- 3 0 6 . . . 制御部

請求の範囲

- [請求項1] ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを生成する伝送ストリーム生成部と、
- 上記伝送ストリームを所定の伝送路を通じて受信側に送信する伝送ストリーム送信部と、
- 上記第1の伝送パケットまたは上記第2の伝送パケットに、上記受信側でデコード時刻および/または表示時刻を取得するための時刻情報を挿入する時刻情報挿入部とを備える
- 送信装置。
- [請求項2] 上記第1の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、
- 上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットにそれぞれ対応したデコード時刻および/または表示時刻を取得するための情報である
- 請求項1に記載の送信装置。
- [請求項3] 上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までのオフセット値とを有する
- 請求項2に記載の送信装置。
- [請求項4] 上記オフセット値は、絶対オフセット値に対応した相対オフセット値であり、
- 上記時刻情報挿入部で挿入される時刻情報には、上記相対オフセット値を上記絶対オフセット値に変換するための変換情報が付加される
- 請求項3に記載の送信装置。
- [請求項5] 上記時刻情報挿入部は、上記相対オフセット値を可変長符号化した

後に挿入する

請求項 4 に記載の送信装置。

[請求項6] 上記時刻情報挿入部で挿入される時刻情報は、上記一つ以上のアクセスユニットのそれぞれに対応した表示時刻、あるいは表示時刻およびデコード時刻である

請求項 2 に記載の送信装置。

[請求項7] 上記第 1 の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記時刻情報挿入部は、上記ペイロードヘッダ部に、上記時刻情報を挿入する

請求項 1 に記載の送信装置。

[請求項8] 上記第 1 の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記ペイロード本体部には、上記伝送メディアがフラグメント化された一つ以上のアクセスユニットをそれぞれ含むフラグメントペイロードがフラグメントヘッダを伴って配置され、

上記時刻情報挿入部は、上記フラグメントヘッダに、対応するアクセスユニットの時刻情報を挿入する

請求項 1 に記載の送信装置。

[請求項9] 上記第 1 の伝送パケットのペイロードは、ペイロードヘッダ部とペイロード本体部とからなり、

上記ペイロード本体部には、上記伝送メディアがフラグメント化された一つ以上のアクセスユニットをそれぞれ含むフラグメントペイロードがフラグメントヘッダを伴って配置され、

上記時刻情報挿入部は、上記フラグメントペイロードに、対応するアクセスユニットの時刻情報を挿入する

請求項 1 に記載の送信装置。

[請求項10] 上記第 2 の伝送パケットのペイロード部は、ペイロードヘッダ部と

ペイロード本体部とからなり、

上記時刻情報挿入部は、上記ペイロード本体部に、上記時刻情報を挿入する

請求項 1 に記載の送信装置。

[請求項11] 上記所定の伝送路は、R F 伝送路または通信ネットワーク伝送路である

請求項 1 に記載の送信装置。

[請求項12] 上記第 1 の伝送パケットおよび上記第 2 の伝送パケットは、MM T パケットである

請求項 1 に記載の送信装置。

[請求項13] ペイロードに伝送メディアを含む第 1 の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第 2 の伝送パケットとが時分割的に多重化された伝送ストリームを生成する伝送ストリーム生成ステップと、

上記伝送ストリームを所定の伝送路を通じて受信側に送信する伝送ストリーム送信ステップと、

上記第 1 の伝送パケットまたは上記第 2 の伝送パケットに、上記受信側でデコード時刻および/または表示時刻を取得するための時刻情報を挿入する時刻情報挿入ステップとを備える

送信方法。

[請求項14] ペイロードに伝送メディアを含む第 1 の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第 2 の伝送パケットとが時分割的に多重化された伝送ストリームを、送信側から所定の伝送路を通じて受信する伝送ストリーム受信部を備え、

上記第 1 の伝送パケットまたは上記第 2 の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、

上記伝送ストリームから取り出された上記伝送メディアを、上記時

刻情報に基づいて取得された上記デコード時刻および/または上記表示時刻を用いて処理する伝送メディア処理部をさらに備える
受信装置。

[請求項15] 上記第1の伝送パケットに含まれる伝送メディアは一つ以上のアクセスユニットで構成され、

上記時刻情報は、上記一つ以上のアクセスユニットの最初のアクセスユニットに対応したデコード時刻あるいは表示時刻の値と、各アクセスユニットにそれぞれ対応したデコード時刻から表示時刻までの絶対オフセット値に対応した相対オフセット値であり、

上記相対オフセット値を上記絶対オフセット値に変換するオフセット値変換部をさらに備える

請求項14に記載の受信装置。

[請求項16] ペイロードに伝送メディアを含む第1の伝送パケットと、ペイロードに上記伝送メディアに関する情報を含む第2の伝送パケットとが時分割的に多重化された伝送ストリームを、送信側から所定の伝送路を通じて受信する伝送ストリーム受信ステップを備え、

上記第1の伝送パケットまたは上記第2の伝送パケットには、デコード時刻および/または表示時刻を取得するための時刻情報が挿入されており、

上記伝送ストリームから取り出された上記伝送メディアを、上記時刻情報に基づいて取得された上記デコード時刻および/または表示時刻を用いて処理する伝送メディア処理ステップをさらに備える

受信方法。