

卒業論文 2002年度 (平成14年度)

MPEG2-TS over IP の設計と実装

慶應義塾大学 環境情報学部

氏名：入野 仁志

指導教員

慶應義塾大学 環境情報学部

村井 純

徳田 英幸

楠本 博之

中村 修

南 政樹

平成15年1月29日

MPEG2-TS over IP の設計と実装

本研究では、民生用 AV 機器を利用した MPEG2-TS over IP システムの設計と実装を行い、既存の同等のシステムと比べて、同等画質で使用帯域の少ないシステムを構築した。

IP ネットワークを通信路とし、映像の入出力に民生用 AV 機器を利用したシステムでは既に DVTS があるが、DVTS は DV フォーマットを利用しているため、使用する帯域が大きくなる。DV フォーマットは圧縮率が一定のため使用するデータ量は不可変である。DVTS はフレーム間引きによって使用帯域を削減可能にしているが、フレーム間引きを利用すると、映像が滑らかではなくなり不自然になる問題がある。

本研究では DV フォーマットより高い圧縮率を持ち、かつ複数の圧縮率をサポートする MPEG2 に注目した。MPEG2 を採用した民生用 AV 機器を利用して、機器から生成される映像データを IP ネットワークを利用して遠隔地へ転送する映像転送システムを設計、実装し、性能評価を行った。このシステムは用途や目的に応じて映像・音声情報の品質を変更可能である。

本研究により、個々の通信環境に応じた品質で、IP ネットワークからの映像・音声のデジタルデータを一般に普及している民生用 AV 機器を用いて、視聴可能となった。

キーワード

1. MPEG2-TS, 2. 民生用 AV 機器, 3. 使用帯域

慶應義塾大学 環境情報学部

入野 仁志

Design and Implementation of MPEG2-TS over IP
with consumer AV equipments

In this thesis, a design and implementation of MPEG2-TS over IP system using consumer AV equipments is described. This system proved that a video stream could be sent through narrower bandwidth without lowering quality of the video.

An existing system, DVTS, is also capable of streaming video and output video to consumer AV equipments. However, streaming DVTS requires vast amount of network bandwidth since the stream is sent in forms of DV format. In DV format, every video frames are compressed equally, and length of the image data is strictly assigned. To adjust bandwidth usage to the narrower lines, DVTS manages to drop frames which minimizes overloaded bandwidth but causes opaque output images.

The solution in this thesis is the use of MPEG2 format. MPEG2 has following characteristics :1)data compression ratio is greater than that of DV format, 2)supports multiple data compression ratio. The design, implementation and the evaluation of the video transport system, which transmit data to distant area over IP network, is shown in this thesis.

Through this thesis, transferring digital video and audio data over IP network using consumer AV equipment has become possible. This system also features manual adjustment of video/audio quality for various usage.

Keywords :

1. MPEG2-TS, 2. consumer AV equipments, 3. bandwidth

目次

第1章	序論	1
1.1	背景	1
1.2	本研究の目的	1
1.3	論文の構成	2
第2章	本研究の技術的要素	3
2.1	民生用デジタル AV 機器	3
2.1.1	インターフェース	3
2.1.2	フォーマット	6
2.2	IP ネットワーク上における映像・音声転送	12
2.2.1	ストリーミング	12
2.2.2	RTP	12
2.2.3	マルチキャスト	13
第3章	既存技術:DVTS	14
3.1	DVTS の特徴	14
3.2	パケットフォーマット	14
3.3	使用帯域の削減	15
3.4	DVTS の問題点	16
第4章	設計	17
4.1	MPEG2-TS 対応機器の利用	17
4.2	設計要件	17
4.3	設計概要	18
4.4	送信部	18
4.4.1	IEEE1394 からのデータ受信	19
4.4.2	IP ネットワークへのデータ送信	20
4.5	受信部	22
4.5.1	IP ネットワークからのデータ受信	22
4.5.2	IEEE1394 へのデータ送信	25
第5章	実装	26
5.1	実装概要	26

5.1.1	実装環境	26
5.1.2	実装概説	26
5.2	送信部	28
5.2.1	IEEE1394 を利用するための準備	28
5.2.2	IP パケット送信	31
5.3	受信部	33
5.3.1	IP パケット受信	34
5.3.2	IEEE1394 への送信	36
第 6 章	評価	39
6.1	本システムの実現した機能	39
6.2	複数の圧縮率への対応	40
6.3	送出データ量の測定	40
6.3.1	測定環境	41
6.3.2	送出データ量の測定	41
6.4	考察	42
第 7 章	結論	44
7.1	まとめ	44
7.2	今後の課題	45

目 次

2.1	DV フォーマット	7
2.2	両方向予測	9
2.3	TS, PES, PS の関係	11
3.1	DV RTP Packet	15
3.2	フレーム間引き	15
3.3	DVTS における画像フレーム間引き率と帯域	16
4.1	本設計の概要図	18
4.2	IEEE1394 Isochronous packet のデータ構造	19
4.3	Source Packet のデータ構造	20
4.4	RTP Header	20
4.5	IEEE1394 Isochronous Header	21
4.6	全データ送信方式の場合の送出されるパケットフォーマット	21
4.7	RFC 準拠方式の場合の送出されるパケットフォーマット	21
4.8	IP パケット受信用バッファから IEEE1394 送信用バッファへのコピー	22
4.9	MPEG2-TS における多重化の仕組み	24
4.10	MPEG2-TS 用の CIP Header	25
4.11	Source Packet Header	25
5.1	Linux における IEEE1394 の取り扱い	27
5.2	ieee1394info 構造体及び send_param 構造体	29
5.3	handle の取得	29
5.4	port の設定と取得	30
5.5	handle への関数の設定	31
5.6	送信側の関数の関係図	32
5.7	関数 <code>_add_to_outbuf()</code> 内での処理の流れ	32
5.8	recv_pmaram 構造体	34
5.9	IP パケット受信用バッファから IEEE1394 送信用バッファへのコピー処理	35
5.10	ieee1394_iso_header 構造体	36
5.11	バッファリングのための <code>usleep</code>	37
5.12	<code>raw1394_iso_write</code> を使った Isochronous 転送	38
6.1	測定時の接続トポロジ	41

6.2	送信側計算機の NIC の送出データ量 (機器の設定:4Mbps)	42
6.3	送信側計算機の NIC の送出データ量 (機器の設定:6Mbps)	42
6.4	送信側計算機の NIC の送出データ量 (機器の設定:12Mbps)	42
6.5	送信側計算機の NIC の送出データ量 (機器の設定:14.4Mbps)	42

表 目 次

2.1	MPEG2 におけるプロファイルとレベルの組合せ	10
2.2	MPEG2 におけるプロファイル@レベルと最大ビットレートの関係	10
5.1	実装ソフトウェア環境	26
5.2	実装に用いた D-VHS デッキ	26
6.1	動作を確認した組合せ (DVHS-DVHS)	39
6.2	動作を確認した組合せ (MICROMV-DVHS)	39
6.3	D-VHS の記録モード	40
6.4	本システム動作状況	40
6.5	測定に用いた送信側計算機	41
6.6	測定の結果 (IPv4)	43
6.7	測定の結果 (IPv6)	43

第1章 序論

1.1 背景

計算機同士の通信はインターネットの登場・普及により IP を利用した通信が一般的になった。また広帯域のデータリンクが普及した。LAN 環境においては IEEE802.3u(100Base-TX)[1] や IEEE802.3ab(1000Base-TX)[2] が普及し各機器は 100Mbps 以上の帯域幅で接続されるようになった。

WAN 環境においてもバックボーンの広帯域化に加えて、ラストワンマイルと呼ばれる一般家庭への接続回線において、ADSL や光ファイバなどのデータリンクの普及により、広帯域かつ常時接続の環境が一般化した。データ量の大きい映像・音声の送受信を行うことが比較的容易になった。一方、広帯域データリンクの登場・普及により、帯域格差はより広がる傾向にある。

BS デジタル放送が 2000 年に国内で開始された。2011 年に現行アナログ放送の打ち切りが予定されており、BS デジタル放送及びその対応機器は、今後さらなる普及が予想される。BS デジタル放送はデータ構造に MPEG2-TS (MPEG2 Transport Stream)[3] を用いている。また対応機器間はデータ送受の為に IEEE1394[4][5] シリアルバスインタフェースを持つ。

民生用 AV 機器がデジタル化し、一部の機器は計算機と共通のインターフェースを持つようになった。これらの機器は計算機と接続して制御やデータの共有・送受が可能である。

1.2 本研究の目的

ネットワークは広帯域化した。また機器のデジタル化により、映像・音声のデジタル情報を特定の通信路で受信し、民生用 AV 機器を利用して視聴する事は可能となった。しかし現時点で IP ネットワークで受けた情報を民生用 AV 機器で視聴するシステムはほぼ皆無である。IP ネットワークを利用した映像転送システムは専用機器で実現されているが、それらの機器は高価である。

本研究の目的は、広帯域化が進む IP ネットワーク環境と、デジタル化が進む民生用 AV 機器の両者の利点を活かすために、IP ネットワーク上での映像転送を、民生用 AV 機器で視聴可能にすることである。また、様々な通信環境での利用を考慮すると、用

途や目的によって、品質を変更可能であることが求められる。

1.3 論文の構成

本論文は第 2 章において、本研究における要素技術として現在の民生用デジタル AV 機器に用いられているインタフェース及びフォーマットに関して述べた後、映像音声情報を IP ネットワーク上において転送される際に利用される技術に関して述べる。第 3 章において関連研究として民生用デジタル AV 機器を利用して映像転送を可能としている DVTS に関して述べ、DVTS の問題点をあげる。第 4 章においてその問題を解決することのできる、本システムの設計に関して述べ、第 5 章で実装に関して述べる。第 6 章で本システムの実装で既存の問題を解決したか否かを評価する。最後に第 7 章で本研究のまとめを行う。

第2章 本研究の技術的要素

本章では，本研究で応用される民生用デジタル AV 機器，及び IP ネットワーク上での映像音声転送に用いられる技術に関して述べる

2.1 民生用デジタル AV 機器

品質の高い映像・音声のデジタルデータは大容量になる．映像・音声のデジタル化は対象の情報をある時間間隔でサンプリングし，その情報を複数の階調に分解して量子化が行われる、そのためより短い時間間隔で，より細かい階調に分ければ品質が高くなるが，それに伴い情報量も増える．音声の場合，サンプリング周波数が時間間隔を表し，量子化ビットレートが階調の細かさを表す映像の場合，フレームレートが単位時間を表し，解像度が 1 画面あたりの細かさを，色数・色解像度が色の階調の細かさを表す．

解像度を $R_x \times R_y$ ，色解像度を n ，フレームレートを f とし，データ量を D_T (単位:bps) とすると，

$$D_t = (R_x R_y) n f$$

となる

例えば TV 程度の画質の映像を表現する場合，TV 程度の画質を解像度が 720×480 ，色解像度が三原色各食 8bit で 24bit，フレームレートが 29.97fps と定義すると，データ量は

$$720 \times 480 \times 24 \times 29.97 \div 1024 \div 1024 = 237.0673828125$$

となり約 237Mbps となる．

このよう映像・音声はデータ量が多い。そのためその転送には，高速インターフェース及び，効率的な圧縮フォーマットが必要となる

本節では，民生用デジタル AV 機器に用いられる、インターフェースとフォーマットに関して述べる．

2.1.1 インターフェース

本項では，計算機と民生用デジタル AV 機器の両者に採用されているインターフェースである IEEE1394 と USB に関して述べる

IEEE1394 や USB は Microsoft 社の PC System Design Guide[6] において必要条件もしくは推奨条件に明記され、その結果現在計算機上で広く普及している。Microsoft PC System Design Guide はハードウェアベンダーが現在最も普及している OS である Microsoft Windows のロゴを取得する為に満たさなければならない条件が記載されている仕様書である。

IEEE1394

IEEE1394 は高速シリアルバスインターフェースで、1995 年に”IEEE1394 Std.1394-1995 IEEE Standard for a High Performance SerialBus” [4] として発表された。その後、拡張が行われ IEEE Std 1394b-2002[5] が 2002 年に規格化された。

IEEE1394 は以下の特徴を持つ。

- 速度
IEEE1394 Std 1394b-2002 で規定されている S3200 で最大で 3.2Gbps の高速通信が可能となる。
- コネクタ形状とケーブルの特徴
IEEE1394 のコネクタは 2 種類あり、6 ピンのコネクタと 4 ピンのコネクタがある。その違いは、電源線の有無である。IEEE1394 の 6 ピンのコネクタの大きさは、約 13.7 × 8.5 × 14.5mm と小さく、4 ピンコネクタはさらに小さい。6 ピンのコネクタは電線は 4 本の信号線と 2 本の電源線の計 6 本で構成されている。ケーブルが細く配線が容易である
- 接続距離と接続トポロジ
IEEE1394 Std 1394b-2002 で規定されているケーブルの最大長は 100m である。最大 16 台の機器を直列に接続可能である。接続トポロジはデジチェーン型とツリー型の接続が可能である。ひとつのバスに 63 台のノードが接続可能でありブリッジの利用により 1023 台まで接続可能である。
- ホットスワップのサポート
IEEE1394 ではホットスワップ (活線挿抜:電源を入れた状態での装置の取り外し) をサポートする。IEEE1394 バスの電源投入時やノードの追加時にバスリセットが起こり、接続トポロジを更新する。接続トポロジが更新される際、機器の識別番号であるノード ID は自律的に決定される。また一つのノードがルートノードとして選択される。このように IEEE1394 アーキテクチャでは自律的に親となるノードを決定し、接続されている機器の管理を行う。

高速通信、小さいコネクタや接続の柔軟性、そして自律的な機器管理により計算機が不要な点から、IEEE1394 は民生用 AV 機器や規格の多くにデジタルデータの入出力

端子として採用されている。

データの転送方式には Isochronous(同期) 転送方式と Asynchronous(非同期) 転送方式がある。

- Isochronous 転送方式
バスが送信可能なデータ量の 80%を占有することが許されており、 125μ 秒毎にデータ転送の優先権が与えられデータ転送を行う事が出来る。データの受信側はデータ到着に対する応答 (Acknowledgement) を返さない。リアルタイム性のデータの転送に用いられる。
- Asynchronous 転送方式
データを定期的に転送することはない転送方式で、データを転送を行う場合は、物理レイヤにバスの制御権を要求し、データの転送を行う。データがブロードキャストデータでない場合はデータの受信側はデータ到着に対する応答 (Acknowledgement) を返す。

USB

USB(Universal Serial Bus) は PC/AT 互換機のレガシーインタフェースを置き換える事を主目的に登場したインタフェースである。1996 年 1 月に USB1.0 が発表され、その後不具合を修正した USB1.1[7] が発表された。また 2000 年 4 月に高速化を図った USB2.0[8] が発表された。

USB には以下の特徴がある。

- 速度
USB は USB2.0 で規定されているハイスピードモードで 480Mbps の通信が可能である。
- コネクタ形状とケーブルの特徴
2 種類のコネクタが規定されている。ホストとなる計算機側とデバイス側でコネクタ形状が違う。計算機側が持つのが A ジャックで、デバイス側が B ジャックとなる。ケーブル側にも対応するように、A プラグと B プラグがある
電線は 2 本の信号線と 2 本の電線で構成されており、ケーブルが細く配線が容易である。
- 接続距離と接続トポロジ
ケーブルの最大長は 5m である。階層型スタートポロジをサポートし、HUB を利用して、最大 127 台のデバイスを接続可能である。HUB は最大 6 階層まで接続可能である。
計算機が機器をコントロールするためのホストコントローラをもち、通信における複雑な処理の殆どを行う。この方法でデバイス側の USB インタフェースを単純化が可能となるが、接続時にはホストコントローラを持つ計算機が基本的には

必要となる。しかし、現在は USB OTG(On-The-GO) 1.0[9] という追加規格があり、計算機を必要としない接続も可能となっている。

- ホットスワップのサポート
USB もホットスワップをサポートしており、計算機及びデバイスの電源が入ったまま、ケーブルの抜き差しが可能である。

基本的に計算機がホストコントローラになることから、計算機の周辺機器としての利用方法を前提とした民生用デジタル AV 機器に採用されている。

データの転送モードはアイソクロナス転送、インタラプト転送、バルク転送、コントロール転送の 4 種類がある

- アイソクロナス転送
リアルタイムのマルチメディアデータなど大容量で一定周期に優先的にデータを転送する時などに利用される。いかなる場合でも最優先で実行され、データの転送幅と転送時間が保証されるがデータのエラー保証は行われない
- インタラプト転送
小容量であるが、一定周期で優先的にデータを転送する場合に利用される。キーボードやマウスなどに用いられる。
- バルク転送
データの発生が不定期ながら大容量のデータを転送するときに用いられる。
- コントロール転送
制御時に用いられる特殊な転送モードである。

IEEE1394 と USB の棲み分け

現時点ではまだ USB 2.0 及び USB OTG が普及段階であり、一般に普及していない。従って現状下では、IEEE1394 が接続トポロジ上に計算機がなくても動作するのに対して、USB は計算機を前提している。また USB 1.1 では 12Mbps のフルスピードモードまでしかサポートされていないため、TV 程度の画質の映像を送ることが難しい。

以上の特徴から IEEE1394 は DV(Digital Video)[10]、D-VHS(Digital VHS)[11][12][13][14]、MICROMV[15]、BS デジタル受信機などの映像機器に採用され、USB は NetMD[16]、デジタルカメラ、web カメラなど、オーディオ、静止画や中・低品質な動画を扱う機器に採用されている。

2.1.2 フォーマット

民生用デジタル AV 機器に使われる映像用フォーマットとして、DV に用いられている DV フォーマット、及び BS デジタル放送等に用いられている MPEG2 に関して述べ

る .

DV フォーマット

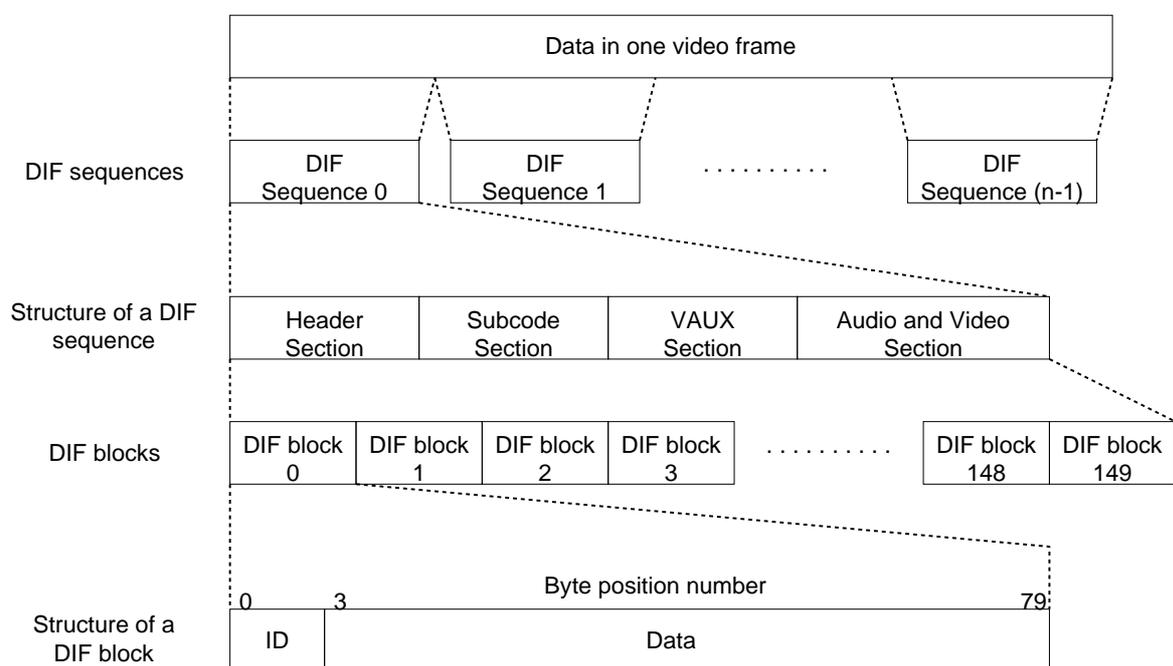
- 圧縮方式

動画は画像を連続して表示することにより実現する . この連続する画像をフレームと呼ぶ . DCT(Discrete Cosine Transform) と VLC(Variable Length Coding) を利用した定レートのフレーム内圧縮を行っている . フレーム内圧縮方式では , 個々のフレーム情報のみで復号できる . 画質は $720 \times 480 \times 29.97\text{fps}$ で , データ量は約 25Mbps となる .

- データ構造

DV フォーマットは , フレーム単位で抽象化されたフォーマットである . 全てのデータ (映像 , 音声 , システムデータ) はビデオフレーム単位で扱われる . フレーム単位で映像と音声を同時に扱う事により , 音声と映像の同期などの問題が解決できる .

DV データは , 3 段階の階層構造で扱われる . DV フォーマットにおける階層構造を図 2.1 に示す .



n = 10 for 525-60 system
n = 12 for 625-50 system

図 2.1: DV フォーマット

ビデオフレームデータは複数の「DIF(Digital Interface Format) sequence」に分割される。DIF sequence は 80byte の DIF block 150 個に分割される。DIF block は、DV データにおける基本構成単位であり、全ての DV 仕様で利用されている。各 DIF block は、3 バイト長の ID ヘッダを含む。ID ヘッダは、DIF block の種類とデータの位置を表す。DV フォーマットでは、DIF block の種類として以下の 5 種類が定義されている。

- ヘッダ
- サブコード
- VAUX(Video Auxiliary Information)
- 音声
- 映像

MPEG2

- 圧縮方式

通信、放送、蓄積メディアなど想定している応用分野が広範囲な MPEG(Moving Picture Expert Group)2[17] は画質に一様な仕様は無い。画面サイズ、フレームレート、色情報の複数の組合せが存在する。画質はプロファイルとレベルの組合せで表す。

表 2.1 にプロファイルとレベルの組合せを示す。表内における空欄はその組合せが存在しないことを示す。表 2.2 プロファイル@レベルとデータ量の関係を示す。映像は両方向予測に基づいたフレーム間圧縮方式を用いている。フレーム間圧縮方式はフレーム差を用いる。

動画像は一般に、あるフレームと次のフレームは近似関係であることが多く、小さなフレーム差を利用することで情報量を減らすことが出来る。MPEG2 のフレーム間圧縮は動き補償の両方向予測によって行われる。

- 動き補償
動きベクトル(動きの方向と大きさ)を検出し、1 フレーム前の信号を検出された動きベクトルの分だけ位置シフトを行う方法である。画像は 16×16 画素のブロックに分割され、各ブロックに対して動きベクトルを検出する。現フレームの当該ブロックにおいて、1 フレーム前のブロックを位置シフトさせ現フレームブロックとの差分量を算出し、その差分量の総和が最小になる位置シフト量を求める。この位置シフト量が動きベクトルである。この方法により、多くのマクロブロックではフレーム差が 0 になる確率が高くなる。
- 両方向予測
予測信号を想定し、その差分を伝送する符号化を予測符号化と言う。動き補

償の場合、動きベクトルで位置シフトした 1 フレーム前のマクロブロックが予測信号となる。

未来 (再生時における時間的に後) の信号から予測を行う方法が両方向予測である。例えば前景の移動により背景が現れた場合、この背景は過去のフレームに存在しない。このような場合に未来の信号から予測を行うことが情報量の削減に有効となる。なお、予測に未来の信号を用いるためにフレームの順序を入れ換え、未来の信号を先に伝送する必要がある。

MPEG では I(Intra) ピクチャ、P(Prediction) ピクチャ、B(Bi-direction) ピクチャの 3 種類のフレームがある。P ピクチャは前方向予測、B ピクチャは両方向予測が用いられる。I ピクチャは予測による誤差を修正するためのフレームで、予測を利用せず、フレーム内で DCT で符号化が行われる。両方向予測を以下の図 2.2 に示す。図中の四角はフレームを表す。四角内の英字はフレームの種類を表し、数字は伝送される順序を表す。また矢印は予測される方向を示す。矢印の始点が予測に利用されるフレーム、矢印の終点が予測が行われたフレームである。例えば I ピクチャはフレーム内で符号化が行われるため、I ピクチャを矢印の終点とした矢印は存在しないが、P ピクチャ・B ピクチャの符号化に必要なため、I ピクチャを始点として、B ピクチャ・P ピクチャを終点とした矢印は存在する。

これらの複数のフレームの集まりを GOP(Group Of Picture) と呼ぶ。GOP 内には必ず I ピクチャが含まれ、ランダムアクセスは GOP 単位で行われる。

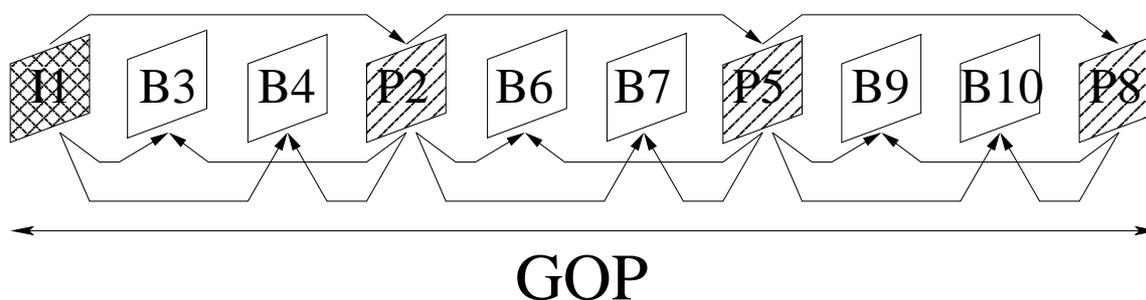


図 2.2: 両方向予測

これらのフレーム間圧縮技術を用いることで、フレーム内圧縮方式よりも圧縮率が高くなる。フレーム内では DCT 及び 2 次元ハフマン符号化を行うことで圧縮率を高める。DV と同等画質の MP@ML は最大ビットレートが DV の約半分である。

- データ構造

映像、音声などの複数のデジタルデータを多重化するときのデータ構造を規格化したものが MPEG システム [3] である

MPEG システムにおいて、映像や音声などコンテンツを構成する要素となる信

表 2.1: MPEG2 におけるプロファイルとレベルの組合せ

High		MP@HL 1920 × 1152 × 60 4:2:0			HP@HL 1920 × 1152 × 60 4:2:2
High-1440		MP@H-14 1440 × 1152 × 60 4:2:0		Spt@H-14 1440 × 1152 × 60 4:2:0	HP@H-14 1440 × 1152 × 60 4:2:2
Main	SP@ML 720 × 576 × 30 4:2:0 B ピクチャなし	MP@ML 720 × 576 × 30 4:2:0	SNR@ML 720 × 576 × 30 4:2:0		HP@ML 720 × 576 × 30 4:2:2
Low		MP@LL 352 × 288 × 30 4:2:0	SNR@LL 352 × 288 × 30 4:2:0		
Level / Profile	Simple	Main	SNR Scalabele	Spatially Scalable	High

表 2.2: MPEG2 におけるプロファイル@レベルと最大ビットレートの関係

profile@level	MAX bitrate [Mbps]
SP@ML	15
MP@LL	4
MP@ML	15
MP@H-14	60
MP@HL	80
HP@ML	20
HP@H14	80
HP@HL	100

号を ES(Elementary Stream) と呼ぶ。ES を可変長のブロックに区切りヘッダ情報を付加したものを PES(Packetized Elementary Stream) と呼ぶ。複数の PES を多重する方法として TS(Transport Stream) と PS(Program Stream) の 2 種類がある。TS と PS は PES 等の要素信号を取り出して相互変換が可能である。TS, PES, PS の関係を図 2.3 に示す。

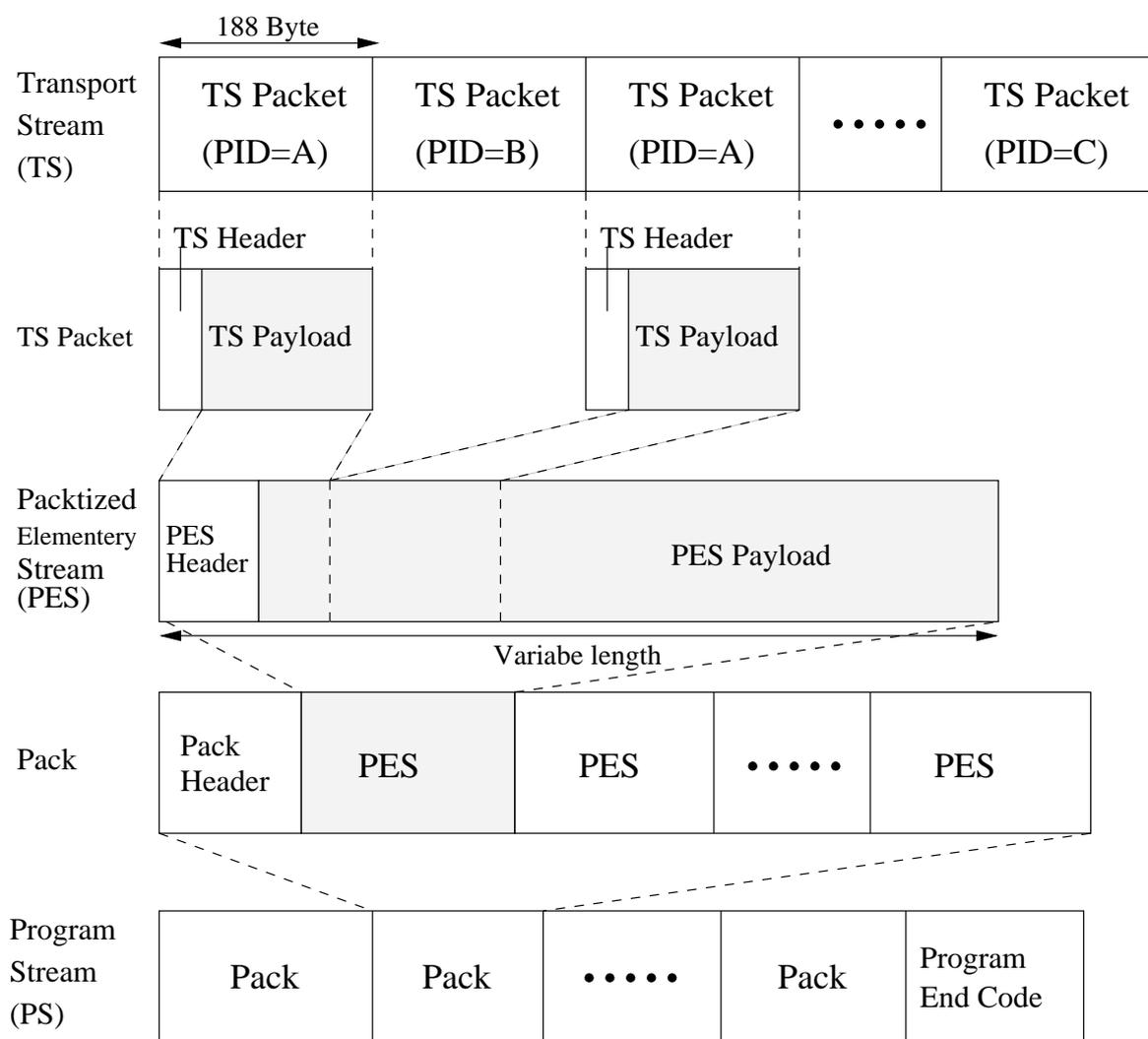


図 2.3: TS , PES , PS の関係

- MPEG2-PS
ローカルディスクからの読みとりなど伝送路誤りが発生しない環境下での利用が想定されている方式である。「パック」と呼ばれる可変長ブロックを単位としたデータ構造を用いている。
民生用デジタル AV1 機器では DVD(Digital Versatile Disk) に用いられている。
- MPEG2-TS
放送や通信といった伝送路誤りが発生しやすい環境下での利用が想定されている方式である。データは 188byte の固定長の MPEG2-TS パケットに分割されて、転送される。この固定長パケットは、固定長のヘッダ、アダプテーションフィールド、ペイロードから構成される。
民生用デジタル AV1 機器では BS デジタル放送受信機 , D-VHS , MICROMV

等に用いられている。

2.2 IP ネットワーク上における映像・音声転送

本項では IP ネットワーク上における映像・音声転送に用いられる技術に関して述べる。

2.2.1 ストリーミング

データ全体を受信してから再生するダウンロード再生に対して、データの受信と並行して逐次再生を行う方法がストリーミング再生である。ストリーミング再生はリアルタイムな映像転送が必要な場合に用いられる。

IP ネットワークを用いた転送ではパケットの到着時間が保証されず、パケット到着時間のジッタ(揺らぎ)が起こる。そのためデータの消費に対してデータの到着が早まる場合や遅れる場合が考えられる。その場合は正しく処理できず、正しく表示、再生されない。

ストリーミング再生を行う場合、データ到着のジッタを吸収するために受信時に一定量のデータを一時的にバッファに蓄積し、蓄積されたデータから消費する方法が取られる。この方法をバッファリングと呼ぶ。

バッファリングはデータ消費をおくらせるため、遅延と大きく関係する。従ってバッファリングを行う量の大きくすると遅延時間も大きくなる。

2.2.2 RTP

実時間に基づく映像音声通信では以下の理由から TCP(Transmission Control Protocol)[18]ではなく UDP(User Datagram Packet)[19] が使われることが多い。

- TCP による輻輳制御が行われるとパケットの到着時間に影響がおり、受信側での再生など実時間に基づいたデータの消費に対して間に合わない可能性が在る。
- TCP によるパケットの再送もパケットの到着時間に影響がある。またデータの消費に対して間に合わなかったパケットを再送しても意味が無い。

UDP を用いると TCP と異なりトランスポートレイヤーにおけるパケットの到着順序は保証されなくなる。パケットの到着順序が正しいか否かを判別するためにはアプリケーションでの対応が必要となる。到着順序を知るための代表的な手法の一例として RTP (Real-time Transport Protocol) [20] がある。

RTP は映像・音声と言った実時間性のあるデータの転送に利用する目的で下位のトランスポートプロトコルとは独立に設計されているプロトコルである。RTP は順序番

号やタイムスタンプ等の情報を付加する．この情報を元に受信側は正しいパケットの順序を知ることが出来る．なお，RTP は配送や配送の到着順序自体を保証するプロトコルではない．

2.2.3 マルチキャスト

複数の宛先に対して同一の内容データの送信を行う場合，ユニキャストを用いた通信では送信元で送出パケットのコピーを行うため，合計のデータ量は宛先の数に比例して増大してしまう．マルチキャストを利用した場合は，経路が分岐する中間ノードにおいて，パケットのコピーを行う．従ってネットワーク上の任意の中間リンクにはコピーは1つしか通らないため，ユニキャストと異なり使用帯域幅が増大しない

そのため同一のデータを任意の複数のホストに一斉配信する場合はマルチキャストを用いるのが一般的である．

第3章 既存技術:DVTS

民生用デジタル AV 機器を用いた既存の研究では DVTS(Digital Video Transport System)[21][22] がある。

3.1 DVTS の特徴

DVTS は DV 機器から IEEE1394 インタフェースを用いて DV フォーマット [23] のデータを読みだし，そのデータに対し IP データグラム化を行い，転送を行うシステムである．受信者はリアルタイムかつ高画質の映像を受信し，鑑賞することが出来る．

DVTS は DV フォーマットを利用するため，以下の DV フォーマットの特徴を持つ．

- 画質はフルフレーム時で $720 \times 480 \times 29.97fps$ であり，圧縮率は一定である．
- 送出データ量はフルフレーム時で 30Mbps 超である．

また DVTS は以下の機能を持つ．

- IPv4/IPv6 に対応する．
- RTP を利用した通信を行う．3.2節で後述する．
- フレーム間引きの機能を持ち，送出データ量を抑えることが可能である．3.3節で後述する．
- multicast の利用により，複数の宛先に対して一斉送信を可能とする．

3.2 パケットフォーマット

DVのためのRTPフォーマット(以後,DV/RTP)[24]を図3.1に示す．DV/RTPフォーマットでは，全てのDVデータは80byteのDIF block単位で扱われる．整数個のDIF blockがDV/RTPパケットに詰められる．DIF blockは，RTPヘッダの直後に連結される．



図 3.1: DV RTP Packet

3.3 使用帯域の削減

フレーム内圧縮である DV フォーマットは画像フレームが全て送信されなくても映像の再生が可能のため DVTS は画像フレームを間引きをサポートしている。この手法により、使用するネットワーク帯域を削減できる。

図 3.2 は、最上段から順に、画像間引きを行わないストリーム、画像間引き率 1/2 のストリーム、画像間引き率 1/3 のストリームを示している。図中の長方形の枠はフレームを表す。最上段の画像間引きを行わない場合に比べて中段の画像間引き率 1/2 のストリームは映像のデータが 2 回に 1 回、最下段の画像間引き率 1/3 のストリームは映像のデータが 3 回に 1 回しか送信されない。DVTS ではフルフレーム時と比較して映像フレームのデータ不足が起きた場合受信側で 1 つ前の映像フレームを最利用する機構を備えており、この方法で受信側でノイズを発生すること無く映像を再生できる。

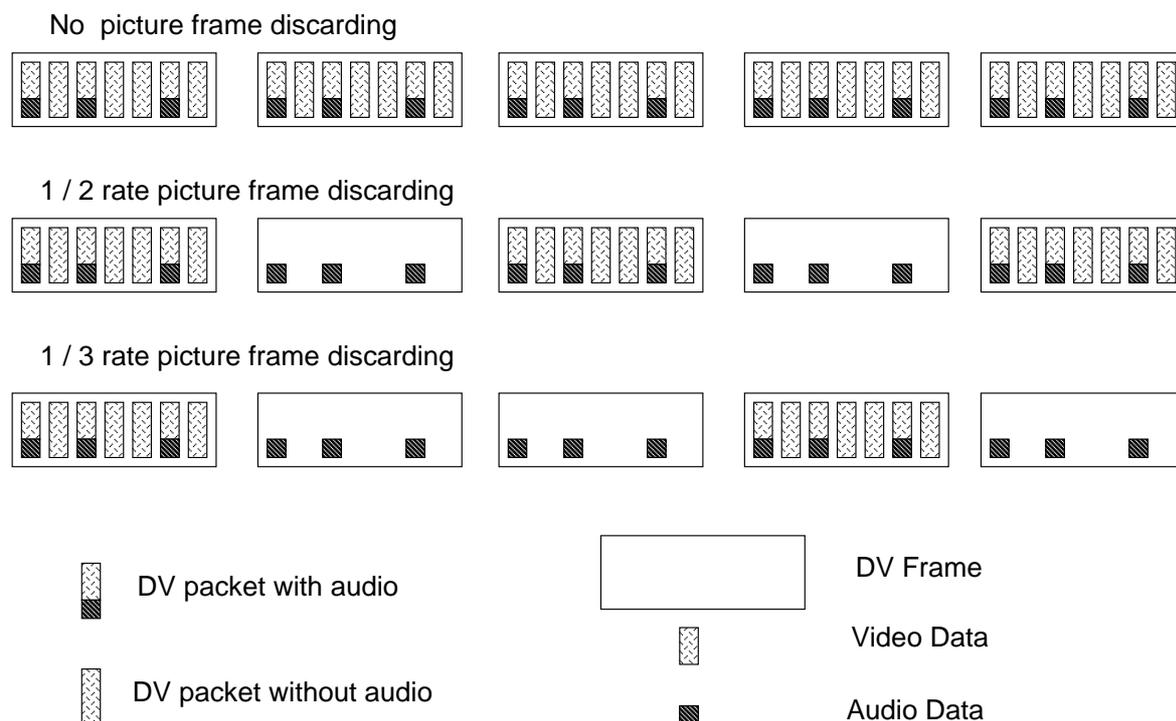


図 3.2: フレーム間引き

図 3.3 に DVTS における画像フレーム間引き率と帯域の関係を示す。フルフレームの場合 30Mbps を超えるデータ量であるが、フレーム数を 10 分の 1 まで間引くと、約

5Mbps のデータ量まで落すことが出来る .

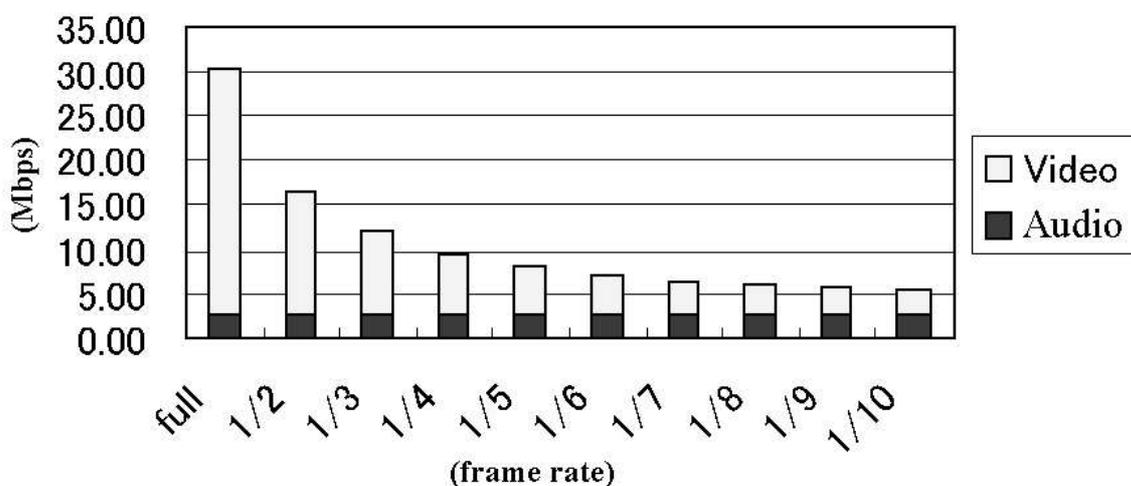


図 3.3: DVTS における画像フレーム間引き率と帯域

3.4 DVTS の問題点

DVTS は DV フォーマットのデータを転送するシステムであり , DV フォーマットは定レートのフレーム内圧縮方式のフォーマットである . そのため DVTS はフレーム内圧縮方式の特徴を持つ .

フレーム内圧縮方式はフレーム毎に独立しているため , 編集用途には好ましい . しかしながら , 同等画質のフレーム間圧縮方式の映像に比べてデータ量は大きく , DVTS の場合フルフレーム時に必要とする帯域が 30Mbps 超と大きい . 使用帯域を削減する方法として , DVTS はフレーム間引きの機能を持つ . DV フォーマットは圧縮率が一定であるため , 画質を変更することが不可能であり , 仕様帯域の削減はフレーム間引きでしか実現できないからである . フレームを間引くと仕様帯域は削減できるが , 映像は滑らかではなくなり , 不自然になる .

複数の圧縮率をサポートしたフレーム間圧縮方式のフォーマットを用いることで , 編集用途には不向きになるが , 鑑賞目的では前述の問題を解決できる .

第4章 設計

4.1 MPEG2-TS 対応機器の利用

3.4節において，DVTS の問題を述べた．以下に3つの問題を整理する．

1. 使用帯域が大きい
2. 圧縮率が一定である
3. フレーム間引きで使用帯域を抑えることができるが，映像が不自然になる

この問題に対する解決方法を以下に示す．

1. DV フォーマットより高い圧縮率を用いるフォーマットを用いる
2. 複数の圧縮率をサポートするフォーマットを用いる

以上から，本研究では，フレーム間圧縮方式でフレーム内圧縮の DV フォーマットよりも高い圧縮率を持ちつつ，複数の圧縮率をサポートし，データ構造としては転送用途に設計されている MPEG2-TS を利用する．MPEG2-TS の利用により高解像度多フレームの高品質映像から低品質の映像まで，目的・用途に応じて画質の変更が可能である．

MPEG2-TS を内部フォーマットとして利用する機器は DV 機器よりも種類が多い．MPEG2-TS を利用する機器は BS デジタル受信機，DVHS，MICROMV があるこれらの機器はインタフェースとして IEEE1394 を採用している．

4.2 設計要件

4.1節より，MPEG2-TS 民生用 AV 機器から送出される MPEG2-TS のデータを，IP ネットワークを利用して遠隔地へ転送して視聴することのできるシステムの構築が本研究における設計要件となる

また 2.2節で述べた IP ネットワーク上での映像音声転送技術及び 3.1節において述べた DVTS の機能を考慮し，以下を本システムの満たすべき機能とする．

- IPv4/IPv6 に対応する

- RTP を利用した通信を行う
- multicast の利用により，複数の宛先に対して一斉送信を可能とする
- 機器が利用できる複数の圧縮率に対応する

4.3 設計概要

本システムは送信部と受信部の 2 つに分かれる．送信側及び受信側において，それぞれ計算機及び MPEG2-TS 機器を利用する．計算機と MPEG2-TS 機器間は IEEE1394 で接続しデータの送受を行う．計算機間は IP を利用してデータの送受を行う．本システムの設計の概要を以下の図 4.1 に示す．

MPEG2-TS 機器から送出されるデータ量はその機器の設定状態によって変わる．一方 IEEE1394 が Isochronous 転送を行う回数は毎秒 8000 回と定まっているため，データ量によって IEEE1394 Isochronous packet のデータ長は変化する．

この変化に完全に対応するために，送信側の機器から送出される IEEE1394 Isochronous Packet をそのまま IP パケット化する方法と，MPEG2-TS Packet のみを抽出して IP パケット化する方法の二つにおいて設計を行った．本研究では前者を 全データ送信方式 と呼ぶ．後者の方法は IP パケット化したときのフォーマットが，MPEG のための RTP ペイロードフォーマットを定めた RFC2250[25] に準拠することから，本研究では RFC 準拠方式 と呼ぶ．

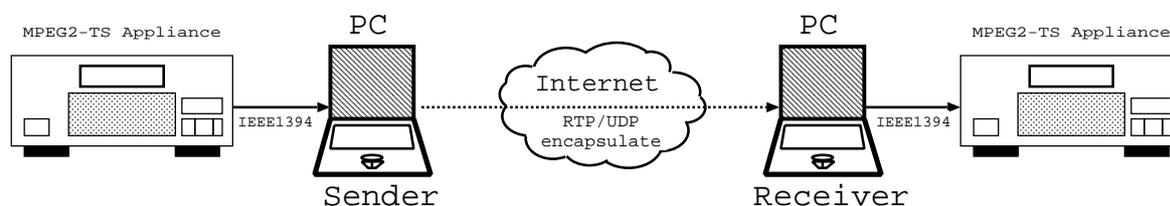


図 4.1: 本設計の概要図

4.4 送信部

送信部分は IEEE1394 からデータを受信する部分と，IP ネットワークに対してデータを送信する部分に分かれる

4.4.1 IEEE1394 からのデータ受信

計算機は MPEG2-TS 機器から IEEE1394 の Isochronous 転送を用いて、IEEE1394 Isochronous packet を受信する。

IEEE1394 Isochronous packet の構造を以下の図 4.2 に示す。

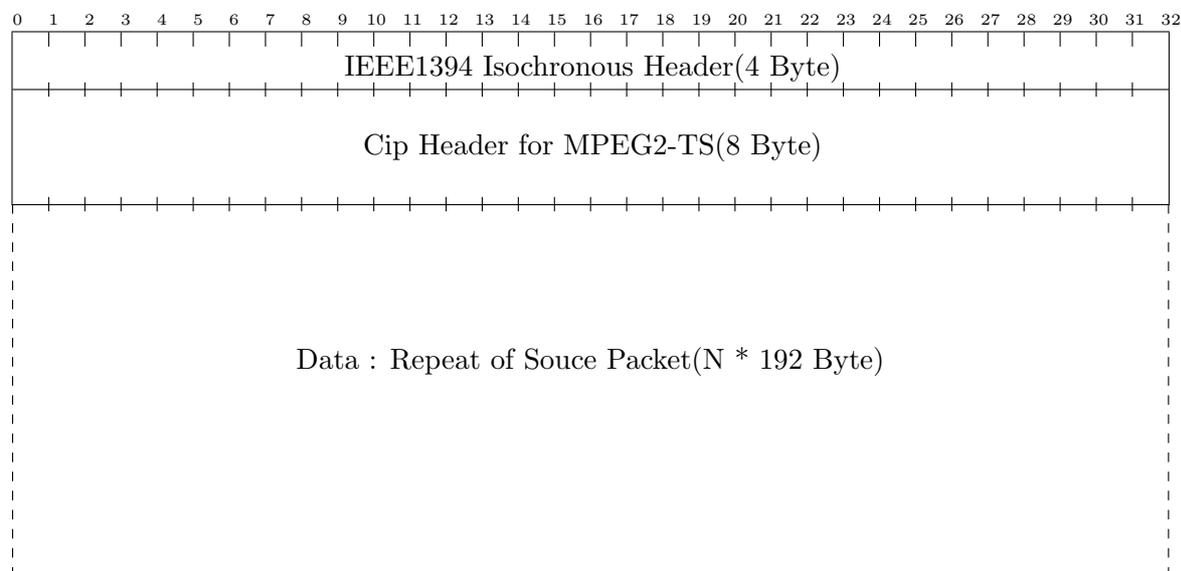


図 4.2: IEEE1394 Isochronous packet のデータ構造

受信する IEEE1394 Isochronous packet は以下の 4 つの部分から構成される。

- IEEE1394 Isochronous Header
IEEE1394 パケットの種類を示す情報が含まれる。
- MPEG2-TS 用の CIP (Common Isochronous Protocol) Header
CIP Header は IEEE1394 上を流れるデータの種類によって規定されている。IEEE1394 上を MPEG2-TS のデータが流れる場合は、MPEG2-TS 用の CIP Header が用いられる。
- Source Packet
Source Packet Header と MPEG2-TS Packet から構成される。IEEE1394 Isochronous packet のデータ部分である Source Packet のデータ構造を以下の図 4.3 に示す。
 - Source Packet Header
IEEE1394 Isochronous ノードの時間情報をもつ。
 - MPEG2-TS Packet
MPEG2-TS のデータである。

Source Packet は 1 つの Isochronous Packet の中に 0 個以上含まれる。そのため、IEEE1394 Isochronous packet のデータ長は可変長である。

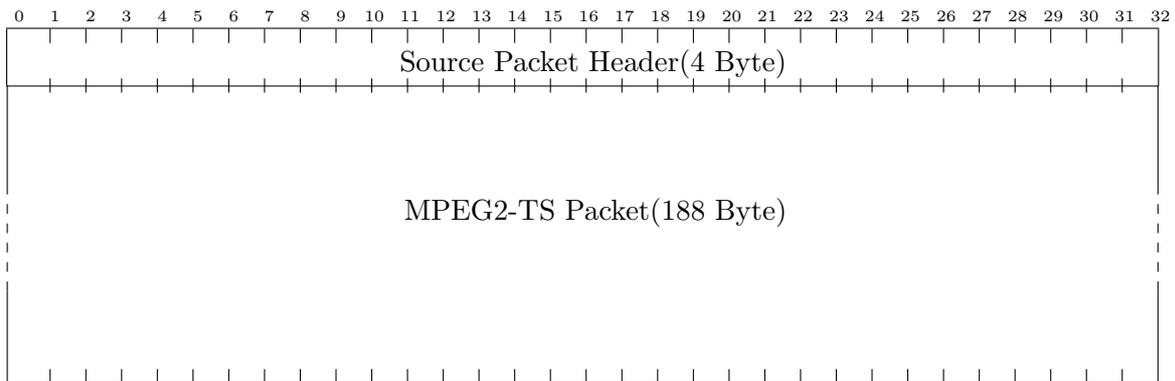


図 4.3: Source Packet のデータ構造

4.4.2 IP ネットワークへのデータ送信

IP ネットワークへのデータ送信を行う部分では，送信用バッファの先頭に RTP ヘッダを付加し，送信用バッファに可能な限りデータを挿入する．RTP ヘッダのフォーマットを以下の図 4.4 に示す．

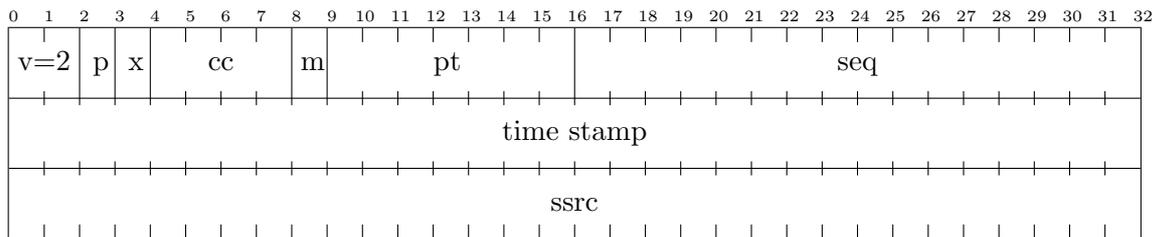


図 4.4: RTP Header

RTP ヘッダのシーケンス番号 (seq) は，IP パケット毎に 1 ずつ増加する．タイムスタンプは 90KHz を単位とする．一方 IEEE1394 Isochronous Packet は 8KHz の周期で機器から送出される，したがって 1 つの IP パケットに挿入される IEEE1394 Isochronous Packet の個数を I とした場合，

$$\frac{I \times 90000}{8000} = \frac{I \times 45}{4}$$

がタイムスタンプの値の増分となる．

全データ送信方式の場合

送信用バッファが溢れないようにデータを挿入するために，送信用バッファにデータを挿入する前に，IEEE1394 Isochronous Header のデータ長を示す data length フィールドの値を調べ，送信用バッファにデータを挿入するか否かを決定する．IEEE1394 Isochronous Header を以下の図 4.5 に示す．

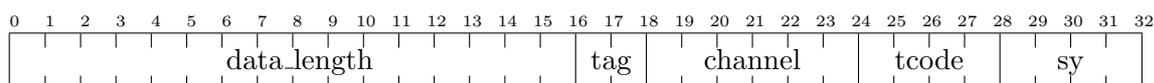


図 4.5: IEEE1394 Isochronous Header

Isochronous Packet の場合パケットの種類を表す tcode の値は 10 となる。また機器から受信する MPEG2-TS Packet を含んだ Isochronous Packet は必ず CIP Header を含む。CIP Header を含む場合は tag の値は 1 になる。IP ネットワークに送出されるパケットフォーマットを以下の図 4.6 に示す。



図 4.6: 全データ送信方式の場合の送出されるパケットフォーマット

RFC 準拠方式の場合

MPEG2-TS Packet は 188byte の固定長のため、送信用バッファに挿入可能な MPEG2-TS Packet の数は一意に決定される。

IEEE1394 Isochronous Packet から MPEG2-TS Packet を抽出するためには、1 つの IEEE1394 Isochronous Packet の中に含まれる MPEG2-TS Packet の数を知る必要がある。この数を N とすると、 N は以下の式で算出できる。

$$\begin{aligned}
 N &= \frac{\text{IEEE1394 Isochronous Packet のデータ長} - \text{CIP ヘッダのヘッダ長}}{\text{Source Packet のパケット長} + \text{MPEG2TS Packet のパケット長}} \\
 &= \frac{\text{IEEE1394 Isochronous Packet のデータ長} - 8}{188 + 4} \\
 &= \frac{\text{IEEE1394 Isochronous Packet のデータ長} - 8}{192}
 \end{aligned}$$

IP ネットワークに送出されるパケットフォーマットを以下の図 4.7 に示す。



図 4.7: RFC 準拠方式の場合の送出されるパケットフォーマット

4.5 受信部

受信部は IP ネットワークからデータを受信する部分と，IEEE1394 ヘデータを送信する部分に分かれる。

バッファは IP パケット受信用のバッファと IEEE1394 への送信用バッファの 2 種類を用意する。IP パケットを受け取った後，受信バッファの内容を IEEE1394 送信用バッファへコピーする。

バッファリングとして IEEE1394 書き込み用バッファに一定量のデータが溜った後は，IEEE1394 への送信と IP パケットの受信を同時並行で行う。

IEEE1394 送信用バッファは Isochronous Header，CIP Header，1 つの Source Packet の合計のデータ長である 204byte 区切りで利用する。

4.5.1 IP ネットワークからのデータ受信

全データ送信方式の場合

受信したデータを IEEE1394 への送信用のバッファにコピーする。受信バッファから IEEE1394 送信用バッファへのコピーを図 4.8 に示す。図中の IEEE1394 Isochronous Packet の後に続く () 内の数字は，その IEEE1394 Isochronous Packet 内に含まれる MPEG2-TS Packet の数をあらわす。処理を単純化するために，IEEE1394 送信用バッファに IEEE1394 Isochronous Packet をコピーする際，その IEEE1394 Isochronous Packet に含まれる MPEG2-TS Packet の数 × 204byte の範囲を利用する。例えば MPEG2-TS が 2 個含まれる Isochronous Packet はデータ長が 396byte であるが，送信用バッファでは 408byte(204byte×2) の範囲を利用し，次の Isochronous Packet とは 12byte の空きを作る。

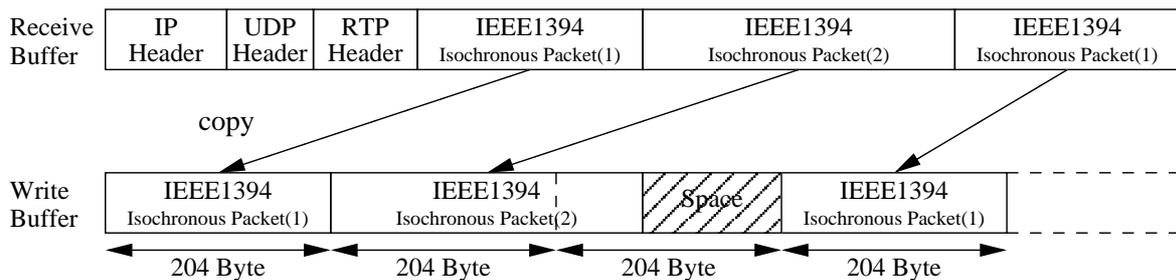


図 4.8: IP パケット受信用バッファから IEEE1394 送信用バッファへのコピー

RFC 準拠方式の場合

受信した MPEG2-TS パケットを元に IEEE1394 Isochronous Packet の再構成し，IEEE1394 送信用のバッファにコピーする。

再構成が必要な部分は IEEE1394 Header , MPEG2-TS 用の CIP Header , Source Packet Header の 3 つである . このうちネットワークからパケットを受信時は IEEE1394 Header , MPEG2-TS 用の CIP Header の 2 つを再構成する . Source Packet Header は機器の時間情報を利用するため , 機器へデータ送信する際に再構成する .

処理は以下の手順で行う

1. 時間当たりのデータ量の算出

1 つの IEEE1394 Isochronous Packet に含む MPEG2-TS Packet の数を決定するために行う . 時間当たりのデータ量を算出するために , MPEG2-TS パケットの PCR(Program Clock Reference) を利用する .

PCR は受信側の MPEG2-TS 機器の STC(System Time Clock) を設定するための時間情報である . ある PCR を PCR_N とし , その次の PCR を PCR_{N+1} とすると PCR_{N+1} と PCR_N 間のデータ量を PCR_{N+1} と PCR_N の値の差分で割ることで時間当たりのデータ量を算出できる .

PCR は特定の MPEG2-TS パケットに含まれる MPEG2-TS では複数の映像 , 音声 , その他の情報を多重化可能な設計になっており , MPEG2-TS パケットに含まれる情報の種類を判別するために MPEG2-TS Header には PID(Packet ID) が含まれている .

PCR は PCR PID をもつ MPEG2-TS パケットに含まれるが , パケットの種類と PID の値の対応は一定ではないため , PCR の値を知るためには PCR PID の値を知る必要がある . パケットの種類と PID の値の対応を行うための対応表も MPEG2-TS のデータの中に含まれる . この対応表は PMT(Program Map Table) と呼ばれる .

PMT のデータが含まれている MPEG2-TS パケットの PID も一定ではなく PAT (Program Assign Table) と呼ばれるパケットの中で定義されている . PAT が含まれるパケットの PID は 0 と規定されている .

MPEG2-TS がこのような階層構造を用いている理由は , 1 つの通信路で複数の番組を送送するために設計されているからである .

PMT が一つの番組内における対応表を表し , PAT が複数の PMT に対する対応表を表すことで 1 つの通信路で複数の番組を送送する事が可能となる .

図 4.9 に MPEG2-TS における多重化と PID の関係を示す .

以上の MPEG2-TS の構造から , PCR を取得するためには以下の 3 つの処理が必要となる .

- (a) PID が 0 のパケットを探し , 解析を行い PMT の PID を取得する .
- (b) PID が PMT の PID と一致するパケットを探し , 解析を行い PCR の PID を取得する .
- (c) PID が PCR の PID と一致するパケットを探し , PCR の値を取得する .

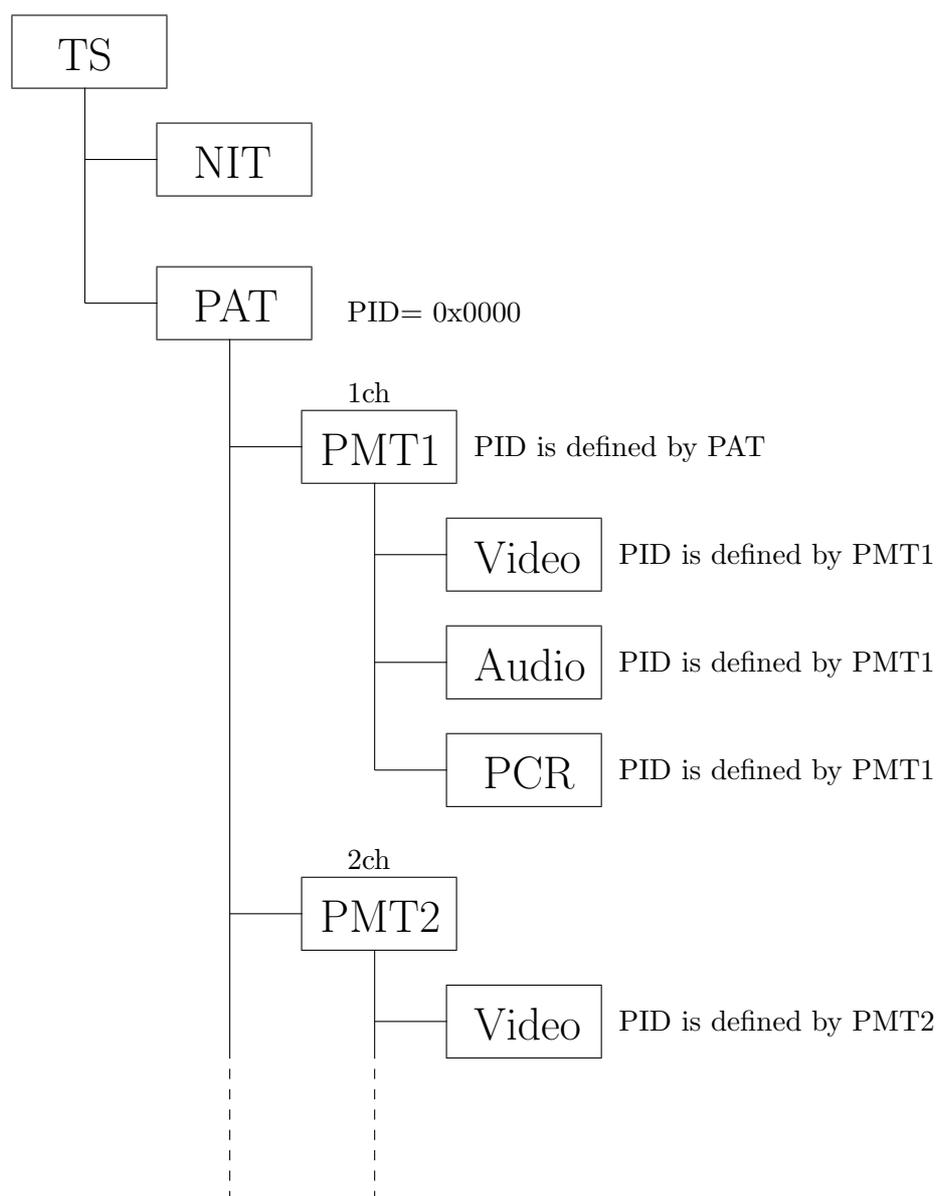


図 4.9: MPEG2-TS における多重化の仕組み

2. データのコピー

算出されたデータ量から，Isochronous Packet のデータ長を決定しそのデータ長に合致するように受信バッファから書き込み用バッファにコピーを行う．全データ送信方式同様，204byte 区切りで書き込み用バッファにコピーを行う．

3. Isochronous Header，CIP Header の再構成

算出されたデータ量から決定された Isochronous Packet のデータ長を元に，Isochronous Header の data length に値を挿入する．また CIP Header を再構成する．CIP Header を図 4.10 に表す．dbc の値は Isochronous 転送の場合，8 の倍数である必要がある．従って各 Isochronous Packet 毎に 8 ずつ増加させる dbc は 8bit で表さ

れるので値が 256 に達した時点で 0 に戻る .

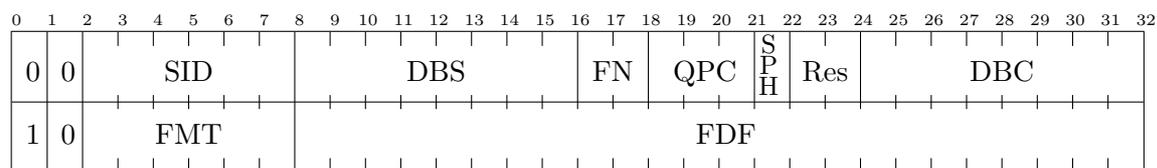


図 4.10: MPEG2-TS 用の CIP Header

4.5.2 IEEE1394 へのデータ送信

全データ送信方式の場合

IEEE1394 への書き込み用のバッファから IEEE1394 の Isochronous 転送を用いて IEEE1394 Isochronous Packet を送信する .

RFC 準拠方式の場合

Source Packet Header を再構成した後 , IEEE1394 への書き込み用のバッファから IEEE1394 の Isochronous 転送を用いて IEEE1394 Isochronous Packet を送信する . Source Packet Header を図 4.11 に示す . Source Packet Header には機器の Cycle Timer Register の下位 25bit の値を挿入する . Cycle Timer Register は Isochronous ノードが持つ時間情報であり , 下位 12bit が 24.576MHz クロックでカウントされ , 次の 13bit が 8KHz クロックでカウントされる .

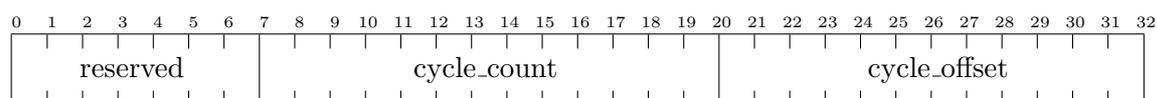


図 4.11: Source Packet Header

第5章 実装

5.1 実装概要

本章では第4章の全データ送信方式の設計に基づき行った実装に関して述べる。

5.1.1 実装環境

表 5.1に実装に用いたソフトウェア環境を示す。

表 5.1: 実装ソフトウェア環境

OS (ディストリビューション)	Linux Usagi 2.4.19 Kernel (gentoo linux 1.4rc1)
IPv6 Stack	Usagi snap (20021111)
必要ライブラリ	libraw1394 0.90
プログラミング言語	C 言語
コンパイラ	gcc 3.2

MPEG2-TS 機器としてはD-VHS デッキを利用した。表 5.2に実装に用いた D-VHS デッキを示す。

表 5.2: 実装に用いた D-VHS デッキ

	メーカー	機種
送信側	日立	DT-DR20000
受信側	Victor	HM-DR10000

5.1.2 実装概説

linux では IEEE1394 に対して複数のアクセス方法を用意している。Linux における IEEE1394 の取り扱いに関して図 5.1に示す。

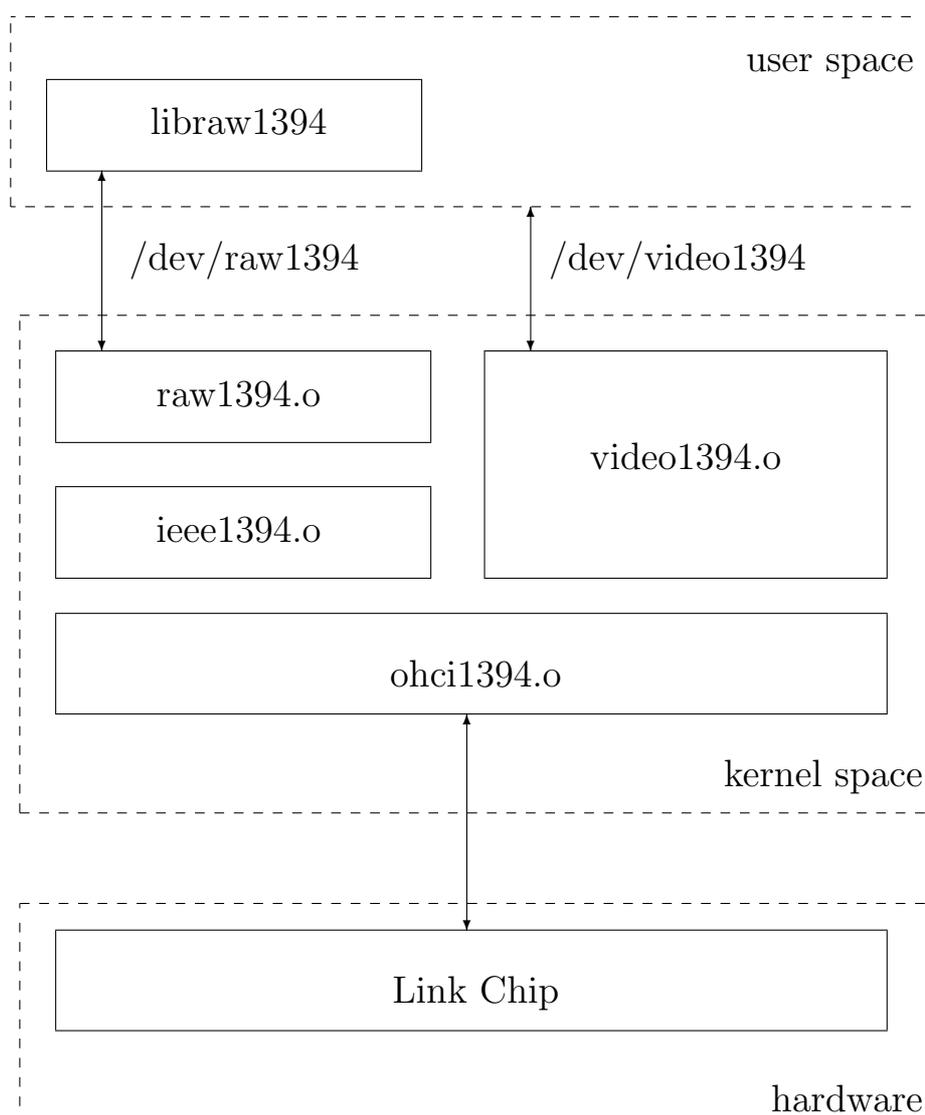


図 5.1: Linux における IEEE1394 の取り扱い

OHCI1394 準拠のインタフェースを実際にアクセスするドライバが `ohci1394.o` である。IEEE1394 を利用する場合はトランザクション管理を行う `ieee1394.o` ドライバを介す、`raw1394.o` ドライバを利用するか、`ohci1394.o` を直接拡張し、`video4linux` 互換インタフェースを提供する `video1394.o` ドライバを利用することになる。ユーザーランドからアクセスするために、それぞれデバイスファイルである `/dev/raw1394`、`/dev/video1394` が存在する。`/dev/raw1394` をユーザーランドから扱う場合は `libraw1394` ライブラリを利用する。

計算機から IEEE1394 機器への Isochronous 転送は、`/dev/raw1394` より `/dev/video1394` が先にサポートしていたため、これまで `/dev/video1394` を使った実装が一般的であった。しかし、`/dev/video1394` は一部に DV に依存した実装が行わ

れているため，本実装では `/dev/video1394` ではなく，`/dev/raw1394` を利用する．

5.2 送信部

本実装送信部の機能を 3 つに分類し以下に概要を示す．

- UDP Socket の作成
UDP Socket を作成するプロトコル独立な関数 `prepare_udp_socket()` を実装した
- IEEE1394 利用のための準備
IEEE1394 を利用するために必要な前処理を行う，`prepare_ieee1394()` を実装した．
5.2.1項で後述する．
- データ送信：IEEE1394 Isochronous Packet の受信と IP ネットワークへのパケットの送信
送信部の処理の中心となる，IEEE1394 Isochronous Packet の受信と，IP ネットワークへのパケットの送信を行う `main_loop()` を実装した．5.2.2項で後述する．

本実装では IEEE1394 からの Isochronous Packet の受信に必要な変数を持つ，`ieee1394info` 構造体と送信に必要な変数を持つ，`send_param` 構造体を独自に定義する．`ieee1394info` 構造体及び `send_param` 構造体を図 5.2 に示す．

`ieee1394info` 構造体では使用する IEEE1394 Isochronous Channel，使用する IEEE1394 のカードの番号，IEEE1394 カードの情報を持つ変数が含まれる．

`send_param` 構造体は `ieee1394` の情報，データの送信先の情報，標準で使うポート，ttl(IPv6 の場合は hop limit)，マルチキャストのための送出インタフェース名を持つ．

5.2.1 IEEE1394 を利用するための準備

linux 上で `libraw1394` を用いて IEEE1394 のを取り扱う場合以下の処理が必要となる．

1. handle の取得
2. port の情報の取得と設定
3. handle への関数の設定

handle の取得

`libraw1394` を用いる場合ファイルディスクリプタの代わりに handle と呼ばれるデータ構造を利用して IEEE1394 機器との通信を行う．handle は具体的には `raw1394handle_t`

```
struct ieee1394info{
    int channel;          /* IEEE1394 isochronous channel */

    raw1394handle_t handle;

    int cardnum;         /* port(card) number */
    struct raw1394_portinfo pinfo[16]; /* port(card) info */
    int is_active;      /* readable flag */
};

struct send_param{
    struct ieee1394info ieee1394info; /* IEEE1394 Parameter */
    struct dest_list_obj *dest_list; /* destination host list */
    struct dest_list_obj *rtcp_out_list; /* rtcp destination list */
    struct rtcp_recv_obj *rtcp_recv_list; /* rtcp src host list */

    int default_port;
    int default_ttl; /* default ttl or hoplimit */
    char *default_ifname; /* interface name for multicast */

    u_int32_t flags; /* flag for configuration */
};
```

図 5.2: ieee1394info 構造体及び send_param 構造体

構造体である。

handle 取得の方法を以下の図 5.3 に示す。handle の取得は関数 `raw1394_new_handle()` を利用する。取得した handle を `send_param` 構造体内に格納する。戻り値が NULL ポインタの場合は失敗である。

```
if( (send_param->ieee1394info.handle = raw1394_new_handle()) == NULL){
    perror("raw1394_new_handle()");
    return(-1);
}
```

図 5.3: handle の取得

port の情報の取得と設定

1 台の計算機には、複数の 1394 バスがある。これらを libraw1394 では port と呼ぶ。通信をする際には、どの port で通信するかあらかじめ指定する必要がある。port の情報の取得と設定の方法を以下の図 5.4 に示す。

```
int numcards;
numcards = raw1394_get_port_info(send_param->ieee1394info.handle,
                                send_param->ieee1394info.pinf, 16);
if(numcards < 0){
    perror("raw1394_get_port_info()");
    return(-1);
} else if (numcards == 0){
    fprintf(stderr, "No IEEE1394 cards found\n");
    return(-1);
} else {
    printf("%d IEEE1394 cards found\n", numcards);
}

if (raw1394_set_port(send_param->ieee1394info.handle,
                    send_param->ieee1394info.cardnum) < 0){
    perror("raw1394_set_port()");
    return(-1);
} else {
    printf("using IEEE1394 card number : %d\n",
           send_param->ieee1394info.cardnum);
}
```

図 5.4: port の設定と取得

port の情報は `raw1394_get_port_info()` を用いて取得する。`raw1394_get_port_info()` の引数には、先に取得した handle、port の情報を持つ pinf 構造体、及び取得したい機器の最大台数を与える。`raw1394_get_port_info()` の戻り値は取得した機器の台数である。また、引数として与えた pinf 構造体に機器の情報が入る。失敗した場合には負の値が返る。また、戻り値が 0 の場合も機器が無い場合、エラーとみなす。

port が存在することを以上の処理で確認した後、利用する port 番号を `raw1394_set_port()` を利用して設定する。`raw1394_set_port()` が失敗した場合戻り値として負の値が返る。利用する port 番号は send_param 構造体内の ieee1394info 構造体内の変数 cardnum が持つ。この変数はあらかじめ main 関数において、値が入力される。

handle への関数の設定

libraw1394 の関数である *raw1394_loop_iterate()* は IEEE1394 上を流れるパケットを受け取った際に任意の関数を呼び出すことができる。対応しているパケットの種類は、Isochronous Packet の他に FCP(Function Control Protocol) Packet などがある。Isochronous Packet を受信した際に *raw1394_loop_iterate()* から呼び出される関数は、関数 *raw1394_set_iso_handler()* を用いて設定する。図 5.5 にその方法を示す。

```
int
prepare_ieee1394(struct send_param *send_param)
{
    (省略)
    /* 関数 _1394_iso_handler を send_param->ieee1394info.handle に設
    定する*/
    raw1394_set_iso_handler(send_param->ieee1394info.handle,
                            send_param->ieee1394info.channel,
                            _1394_iso_handler);
}
```

図 5.5: handle への関数の設定

raw1394_set_iso_handler() は引数に handle と関数のポインタをとる。*raw1394_set_iso_handler()* によって、指定した handle 上で Isochronous Packet が流れるたびに指定した関数を呼び出すことができる。Isochronous Channel には全ノードからの入力を受け付ける、ブロードキャストチャネルである 63 を用いる。

5.2.2 IP パケット送信

raw1394_loop_iterate() を利用して、IEEE1394 Isochronous Packet を受信するたびに 5.2.1 項において設定した関数が呼ばれる。図 5.6 に関数の関係図を表す。

IEEE1394 Isochronous Packet 受信時に *raw1394_loop_iterate()* から呼び出される、*_1394_iso_handler()* は受信した IEEE1394 Isochronous Packet のデータ長を調べ、正しいデータ長の場合は *add_to_outbuf()* を呼び出す。*add_to_outbuf()* 内で行う処理は以下の通りである。図 5.7 に流れ図で示す。

- 送信バッファに追加可能か調べる
受信した Isochronous Packet のデータ長からバッファに追加可能か判断する。不可能な場合は以下の処理を行う。

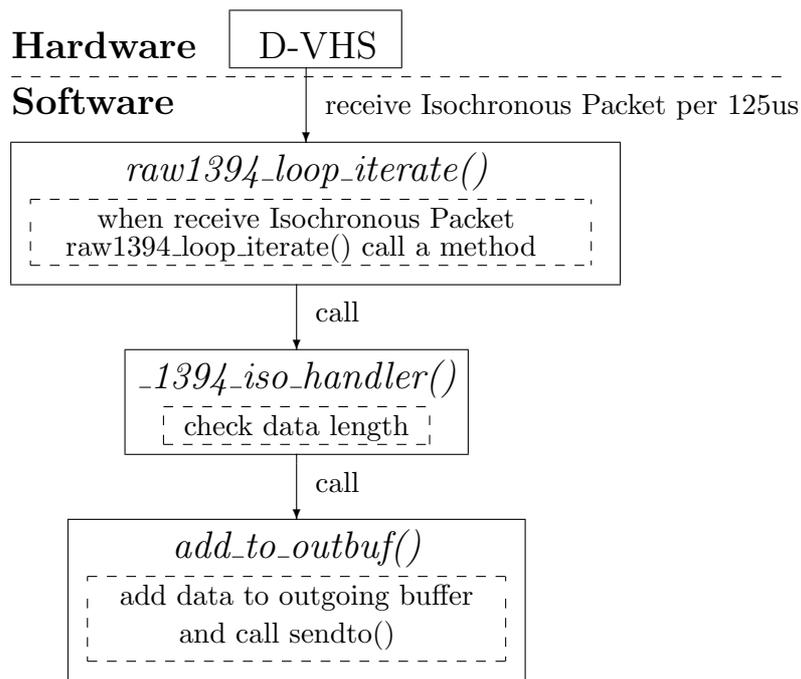


図 5.6: 送信側の関数の関係図

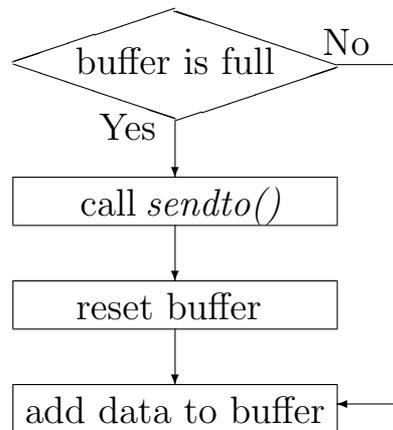


図 5.7: 関数 `_add_to_outbuf()` 内での処理の流れ

- パケットの送信
`sendto()` システムコールを呼び出す .
- バッファの初期化
バッファの追加位置を示す変数を 0 に戻し次の送信時のために , 1 パケット内に挿入した Isochronous Packet を記録した変数を元に RTP のタイムスタンプを算出し , 増加させる . また RTP の連続番号 (seq) を 1 増加する 1 パケット内に含まれる Isochronous Packet を記録する変数を 0 にする .

- 受信した IEEE1394 Isochronous Packet をバッファに追加する
memcpy() でデータをコピーし、バッファの追加位置を示す変数を追加したデータ長だけ増やし、1 パケット内に含まれる Isochronous Packet を記録する変数を 1 増やす。

5.3 受信部

本実装受信部の機能を 5 つに分類し以下に概要を示す。

- UDP Socket の作成
UDP Socket を作成し、プロトコル独立な関数 *prepare_udp_socket()* を実装した。
- IEEE1394 利用のための準備
IEEE1394 を利用するために必要な前処理を行う、*prepare_ieee1394()* を実装した。本処理は 5.2.1 項で述べた処理とほぼ同等で、port の設定までを行う。
- pthread の利用
IEEE1394 への Isochronous Packet の送信と、IP ネットワークからのパケット受信を、同時並行に行うために pthread を利用する。*pthread_create()* と *pthread_join()* を行う *prepare_pthread()* を実装した。
- IP パケット受信部分
IP ネットワークからのパケット受信を行い、後述する IEEE1394 への送信用のバッファにデータをコピーする、*rtp_read_loop()* を実装した。5.3.1 項に後述する。
- IEEE1394 への送信部分
IEEE1394 へのパケット送信を行う、*ieee1394_write_loop* を実装した。5.3.2 項に後述する。

本実装では、独自に定義した、*recv_param* 構造体を利用する。*recv_param* 構造体を図 5.8 に示す。

recv_param 構造体では、受信した IP パケットからの IEEE1394 Isochronous Packet 読み込み回数を数える変数群と、MPEG2-TS 機器への IEEE1394 Isochronous Packet 書き出し回数を数える変数群を持つ。IP パケット受信部分と IEEE1394 への送信部分でそれぞれ読み込み回数変数と書き出し回数を変化させる。これらの値の比較でバッファリングを行う。

本実装では、バッファを 2 つ利用する。IP ネットワークから受信する IP パケットのデータを保持する受信用バッファと MPEG2-TS 機器に対して IEEE1394 への送信処理に必要なデータを保持する、送信用バッファである。

recv_param 構造体内の *video_buf* は書き込み用バッファの先頭ポインタである。書き込み用バッファは、*main* 関数において、*malloc()* を利用してメモリ領域を確保する。

```
#define IFNAMSIZ      16

struct recv_param {
    struct ieee1394info ieee1394info;    /* IEEE1394 Parameter */

    char *multicast_addr_str;    /* Strings for Multicast Address */
    /* Interface Name for Multicast */
    char multicast_ifname[IFNAMSIZ];
    char *video_buf;    /* buffer for writeing IEEE1394 */
    /* pthread for write thread and read thrad */
    pthread_t write_thread, read_thread;
    /* counter for read mpeg2-ts from ip packet */
    u_int32_t readcount, maxreadcount, lastreadcount;
    /* counter for write to 1394 iso packet */
    u_int32_t writecount;
    u_int32_t readcounttotal;    /* count total of readcount */
    u_int32_t writecounttotal;    /* count total of writecount */
}
```

図 5.8: recv_pmaram 構造体

5.3.1 IP パケット受信

受信した IP パケットは RTP ヘッダと、IEEE1394 Isochronous Packet から構成されている。RTP ヘッダを取り除き、後続する IEEE1394 Isochronous Packet を 1 つずつ、IEEE1394 送信用バッファにデータのコピーを行う。データのコピーをする際、4.5.1 項の図 4.8 に示した通り、送信用バッファ 204byte 毎に行う。図 5.9 にバッファのコピー処理を示す。

バッファのコピー処理は IEEE1394 Isochronous Packet のデータ長ごとに行う。IEEE1394 Isochronous Packet のデータ長を取得するために、図 4.5 に示した IEEE1394 Isochronous Header の構造に準じて `ieee1394_iso_header` 構造体を定義する。図 5.10 に `ieee1394_iso_header` 構造体を示す。

IEEE1394 Isochronous Packet のデータ長を取得するために、`ieee1394_iso_header` 構造体のポインタにコピーすべきデータの先頭ポインタを代入する。`ieee1394_iso_header` 構造体内には IEEE1394 Isochronous Packet のデータ長を示す、変数 `len` が定義されている。

以上の処理で取得した IEEE1394 Isochronous Packet のデータ長から、その IEEE1394 Isochronous Packet 内に含まれる MPEG2-TS パケットの数を取得する。MPEG2-TS の

```
#define                M2TSISOPKTSIZ    204

u_int16_t readpoint;
u_int8_t add_readcount; /* temporary counter for readcount */
struct ieee1394_iso_header *isohdr_pt;

for(readpoint=0; readpoint < recvlen - sizeof(rtp_hdr_t); ){
    isohdr_pt = (struct ieee1394_iso_header *)
                &recvbuf[sizeof(rtp_hdr_t) + readpoint];

    /* count MPEG2-TS packet in UDP packet */
    add_readcount = ntohs(isohdr_pt->len) / 192;
    if(add_readcount == 0){ /* if packet is empty */
        add_readcount = 1;
    }
    memcpy(recv_param->video_buf +
           recv_param->readcount * M2TSISOPKTSIZ,
           &recvbuf[sizeof(rtp_hdr_t) + readpoint],
           ntohs(isohdr_pt->len) + 4);
    recv_param->readcount += add_readcount;
    recv_param->readcounttotal += add_readcount;
}
```

図 5.9: IP パケット受信用バッファから IEEE1394 送信用バッファへのコピー処理

パケットの数は IEEE1394 Isochronous Packet のデータ長を、Source Packet(図 4.3 参照) のデータ長である 192 で割って算出する。変数 `add_readcount` は新たにコピーする MPEG2-TS パケットの数を保持すると同時に IEEE1394 送信用バッファの利用する範囲を決定するための変数である。機器の設定状態によっては Source Packet を 1 つも含まない IEEE1394 Isochronous Packet(以後、空パケット) が含まれる場合がある。その場合は変数 `add_readcount` を 1 にして、IEEE1394 送信用バッファに Isochronous 転送 1 回分の空きの領域を作る。空パケットは IEEE1394 上で機器がバッファオーバーフローを起こさないようにするために必要な情報である。算出された変数 `add_readcount` の値をもとに、`memcpy()` でバッファのコピーを行い、MPEG2-TS パケットの読み込み回数を数える変数である `recv_param` 構造体内の `readcount` 及び `readcounttotal` を `add_readcount` の値だけ増加させる。

```
/*
 *IEEE1394 Isochronous Header
 */

struct ieee1394_iso_header{
    u_int16_t    len;          /* data length */
#ifdef BYTE_ORDER == BIG_ENDIAN
    u_int8_t     tag:2;        /* Isochronous Data Tag */
    u_int8_t     channel:6;    /* Isochronous Channel */
    u_int8_t     tcode:4;      /* Transaction Code */
    u_int8_t     sy:4;         /* Synchronize Code */
#endif
#ifdef BYTE_ORDER == LITTLE_ENDIAN
    u_int8_t     channel:6;    /* Isochronous Channel */
    u_int8_t     tag:2;        /* Isochronous Data Tag */
    u_int8_t     sy:4;         /* Synchronize Code */
    u_int8_t     tcode:4;      /* Transaction Code */
#endif
};
```

図 5.10: ieee1394_iso_header 構造体

5.3.2 IEEE1394 への送信

バッファリング

pthread を利用しているため、IP ネットワークからのパケット受信が、IEEE1394 書き込み処理より先に行われる保証がない。そこで、書き込み用バッファへ一定量データがたまるまで、IEEE1394 書き込み処理を止める必要がある。本実装では、一度空になった場合、即ち読み出し回数を数える変数である readcount, readcounttotal がそれぞれ書き出し回数を数える変数である writecount, writecounttotal と等しくなった場合に、書き込み用バッファサイズの半分がたまるまで停止する。5.11 にバッファリングを示す。

IEEE1394 への送信

送信を行うためには送信すべき IEEE1394 Isochronous Packet のパケット長を知る必要がある。5.3.1 項で述べた通り、IEEE1394 Isochronous Packet のパケット長は

```
if(emptyflag){
    if(recv_param->readcount - recv_param->writecount <
        recv_param->maxreadcount / 2){
        usleep(125 * recv_param->maxreadcount / 2);
    }
    emptyflag = !emptyflag;
}
```

図 5.11: バッファリングのための `usleep`

`ieee1394_iso_header` 構造体のポインタを利用して取得する。

機器への Isochronous 転送は `raw1394_iso_write()` を利用して行う。図 5.12 を使った Isochronous 転送の方法を表す。

関数 `raw1394_iso_write()` は図 4.5 に示した IEEE1394 Isochronous Header の各フィールドに値を代入することができる。Isochronous Channel には全ノードが反応するブロードキャストチャンネルである 63 を用いる。tcode は CIP ヘッダを含むため 1 となり、sy は 0 となる。また IEEE1394 のスピードはどの機器でも動作するように、100Mbps に設定した。関数 `raw1394_iso_write()` は引数から IEEE1394 Isochronous Header を生成し、第 7 引数であるポインタから始まるデータを連結し IEEE1394 Isochronous Packet を生成し、指定したスピードで送信を行う。書き込み用バッファには IEEE1394 Isochronous Header も含めてデータが保持されている。従って第 7 引数のポインタには IEEE1394 Isochronous Header の 4byte 分を含まず CIP ヘッダの先頭アドレスを渡す。関数 `raw1394_iso_write()` は失敗すると返り値に -1 未満の値を返す。

```
/* prototype declaration in rwa1394.h */
/*
int raw1394_iso_write(raw1394handle_t handle, unsigned int channel,
                    unsigned int tag, unsigned int sy,
                    unsigned int speed, size_t length, quadlet_t *data);
*/

#define DEFAULT_ISO_SEND_CHANNEL 63

struct raw1394_handle_t *handle;
struct ieee1394_iso_header *isohdr_pt;

if(raw1394_iso_write(recv_param->ieee1394info.handle,
                    DEFAULT_ISO_SEND_CHANNEL, 1, 0,
                    L1394_SPEED_100, ntohs(isohdr_pt->len),
                    (quadlet_t *)
                    &recv_param->video_buf
                    [sizeof(struct ieee1394_iso_header) +
                    recv_param->writecount * 204]
                    ) < -1)

{
    perror("raw1394_iso_write");
}
```

図 5.12: raw1394_iso_write を使った Isochronous 転送

第6章 評価

本章において本実装の評価に関して述べる．

6.1 本システムの実現した機能

本システムでは MPEG2-TS フォーマットの映像・音声を IP ネットワークを用いて、転送することを実現した．満たした機能は以下の通りである．

- IPv4/IPv6 に対応した．
- RTP を利用した通信を行った
- 機器が選択することの出来る複数の圧縮率に対応した．
6.2に後述する．

また、以下の2つの機器の組合せでの動作を確認した．動作した機器の組合せを以下の表 6.1、表 6.2 に示す．

表 6.1: 動作を確認した組合せ (DVHS-DVHS)

	種類	メーカー	機種
送信側	D-VHS	日立	DT-DR20000
受信側	D-VHS	Victor	HM-DR10000

表 6.2: 動作を確認した組合せ (MICROMV-DVHS)

	種類	メーカー	機種
送信側	MICROMV	SONY	DCR-IP7
受信側	D-VHS	Victor	HM-DR10000

6.2 複数の圧縮率への対応

表 6.1 に示した機器の組合せにおいて送信側に利用している、日立の D-VHS デッキである DT-DR20000 は IEEE1394 に送出するデータ量を 4Mbps, 6Mbps, 12Mbps, 14.4Mbps と段階的に変更することができる。機器の設定をどの状態に変更しても解像度、フレームレートは変わらない。D-VHS のデータ量と対応する画像フォーマットは表 6.3 に示す関係で規定されている。[11]

表 6.3: D-VHS の記録モード

記録モード	最大データ量	対応画像フォーマット
HS	28.2Mbps	1080i/720p
STD	14.1Mbps	480p/480i
LS2	7.0Mbps	480i
LS3	4.7Mbps	480i
LS5	2.8Mbps	480i
LS7	2.0Mbps	480i

左列の記録モードは上段から順に、高画質モード、標準モード、2倍長時間モード、3倍長時間モード、5倍長時間モード、7倍長時間モードを示す。右列の対応画像フォーマット内の i はインタレース、p はプログレッシブを表し、数字は走査線を表す。本システムは、送信側の機器の設定が 4Mbps, 6Mbps, 12Mbps, 14.4Mbps の全ての場合において送信したデータを受信側の機器で表示することが出来た。以上から、本システムは HS, STD, LS2, LS3 の規格に対応した。本システムの動作状況を表 6.4 に示す。

表 6.4: 本システム動作状況

記録モード	HS	STD	LS2	LS3
DT-DR20000 上の設定	14.4Mbps	12Mbps	6Mbps	4Mbps
動作状況				

6.3 送出データ量の測定

本節において、本実装の送出データ量を測定し、使用帯域を低く抑えた状態で滑らかな映像の転送が可能か評価する。

6.3.1 測定環境

送信側，受信側にそれぞれ計算機と D-VHS を用意し，計算機と D-VHS を IEEE1394 ケーブルで接続し，計算機間はスイッチを介して UTP で接続した．測定時の接続トポロジを図 6.1 に示す．

測定に用いた計算機のソフトウェア環境は実装環境と同一である (表 5.1)．測定に用いた送信側の計算機のハードウェア環境を表 6.5 に示す．

表 6.5: 測定に用いた送信側計算機

CPU	PentiumIII-S 1.26GHz
メモリ	256MB
NIC	100Base-TX(Intel EtherExpressPro100 内臓)

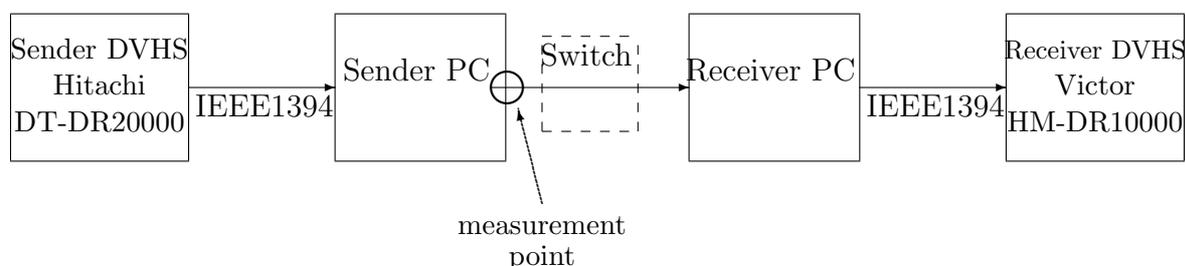


図 6.1: 測定時の接続トポロジ

6.3.2 送出データ量の測定

送信側のデスクで送出データ量を変更しそれぞれの場合において送信側の計算機のネットワークインタフェースの送出データ量を 10 分間、1 秒毎に測定した．

送信側 D-VHS デッキの送出データ量をそれぞれ 4Mbps , 6Mbps , 12Mbps , 14.4Mbps に変更した際の計測結果を図 6.2 , 図 6.3 , 図 6.4 , 図 6.5 に示す．

以上の測定の結果を表 6.6 及び表 6.7 に示す．6.6 は IPv4 での通信時，6.7 は IPv6 での通信時の結果である．表 6.6 は上段から，機器の設定状態，本測定の平均値，最大値，最小値，標準偏差，理論値での MPEG2-TS データ量，オーバーヘッドとしている．ここでのオーバーヘッドとは，実測値を理論値で割った値を示している． 図 6.2 , 図 6.3 , 図 6.4 , 図 6.5 の結果及び表 6.6 , 表 6.7 の標準偏差の値の小ささから，送信側の D-VHS デッキの送出データ量がどの場合でも，送信側計算機は若干の揺らぎがあるもののほぼ一定のデータを送出していることがわかる．

本実装は IP ネットワークデータを送出する際，IEEE1394 Isochronous Packet 全体

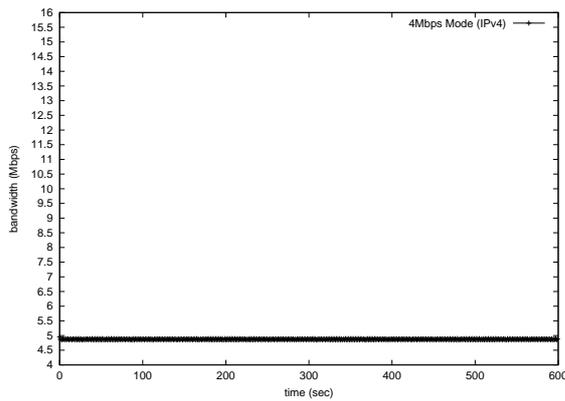


図 6.2: 送信側計算機の NIC の送出データ量 (機器の設定:4Mbps)

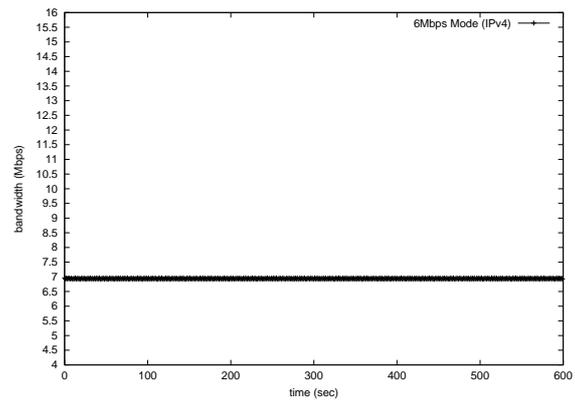


図 6.3: 送信側計算機の NIC の送出データ量 (機器の設定:6Mbps)

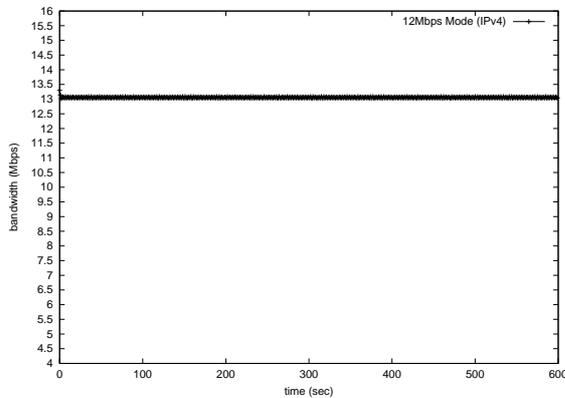


図 6.4: 送信側計算機の NIC の送出データ量 (機器の設定:12Mbps)

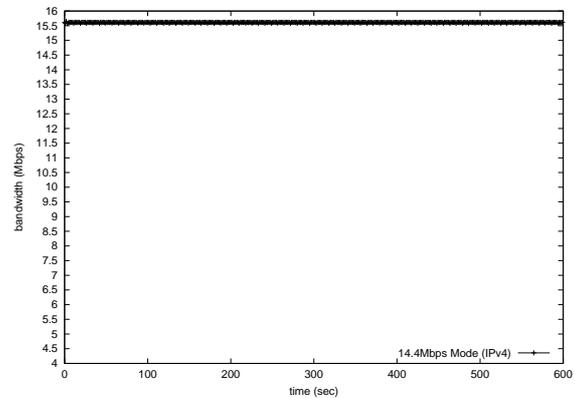


図 6.5: 送信側計算機の NIC の送出データ量 (機器の設定:14.4Mbps)

を送出している．この方法で送信側のデータ転送のタイミングを受信側において，正確に再現できる一方，映像復号化に必要なデータ外のデータを送ることになり，使用する帯域が大きくなる．

機器の設定が小さいデータ量に設定されているほど，IEEE1394 Isochronous Packet の空パケットが増えるため，オーバーヘッドが大きくなる結果となった．

6.4 考察

送信側の D-VHS デッキの送出データ量がどの場合でも，解像度と 1 秒あたりのフレーム数は変化しない．このことから，DVTS の問題であった使用帯域を低く抑える場合はフレームを間引かなければならないという問題を解決することができた．

圧縮率は 1GOP 内に含まれるフレーム数で変化する．1GOP 内に含まれるフレーム数を増やし，予測を行わないフレームである I ピクチャに対して，予測を行うフレームである P ピクチャ，B ピクチャの数を増やすことで圧縮率を高めることができる．符

表 6.6: 測定の結果 (IPv4)

機器の設定	4Mbps	6Mbps	12Mbps	14.4Mbps
測定平均値	4.87Mbps	6.94Mbps	13.06Mbps	15.61Mbps
最大値	4.95Mbps	6.96Mbps	13.30Mbps	15.65Mbps
最小値	4.85Mbps	6.92Mbps	13.02Mbps	15.58Mbps
標準偏差	0.02	0.02	0.02	0.02
MPEG2-TS のデータ量 (理論値)	3.82Mbps	5.74Mbps	11.47Mbps	13.77Mbps
オーバーヘッド	1.274	1.210	1.138	1.134

表 6.7: 測定の結果 (IPv6)

機器の設定	4Mbps	6Mbps	12Mbps	14.4Mbps
測定平均値	4.94Mbps	7.04Mbps	13.23Mbps	15.85Mbps
最大値	4.99Mbps	7.07Mbps	13.26Mbps	15.89Mbps
最小値	4.92Mbps	7.03Mbps	13.22Mbps	15.59Mbps
標準偏差	0.02	0.02	0.02	0.02
MPEG2-TS のデータ量 (理論値)	3.82Mbps	5.74Mbps	11.47Mbps	13.77Mbps
オーバーヘッド	1.292	1.227	1.153	1.151

号化，復号化は GOP 単位で行われるため，圧縮率が高い程，遅延が大きくなる．実際に機器の設定を 14.4Mbps に設定した場合に比べて，4Mbps に設定した場合は，遅延が大きくなることが確認された．

第7章 結論

7.1 まとめ

本研究は、MPEG2-TS 対応の民生用 AV 機器を用いて、IP ネットワークを通進路とした映像転送システムの設計と実装を行った。本システムは通信環境に応じて映像・音声情報の品質を変更することが出来る。

民生用 AV 機器を用いて、IP ネットワークを通信とする既存の映像転送システムは、DVTS がある。しかしながら DVTS は DV フォーマットに依存している。そのため使用帯域が大きく、また圧縮率を変更することが出来ない。DVTS において、使用帯域を減少させるためには、フレーム間引きを行う必要があった。しかしフレーム間引きを行うと、映像が不自然になる問題があった。この問題に対して本研究ではフレーム間圧縮方式のフォーマットであり、複数の圧縮率をサポートする、MPEG2-TS に対応した民生用 AV 機器を利用した。

MPEG2-TS は圧縮率によって使用するデータ量が変化するが、MPEG2-TS 対応民生用 AV 機器のインタフェースに用いられている IEEE1394 の Isochronous(同期) 転送は毎秒 8000 回の一定速度で行われるため、Isochronous Packet のデータ長はパケット毎に変化する。

この変化に対応するために、本研究では2つの設計を行った。1)RFC 準拠方式、2)全データ送信方式である。前者は、送信側は RFC2250 に規定されているパケットフォーマットで IP ネットワークにパケットを送信する方法である。Isochronous Packet に含まれる MPEG2-TS パケットのみを抽出して送信し、受信側は MPEG2-TS のデータ内に含まれる時間情報の一つである PCR を利用して、Isochronous Packet データ長を算出し、Isochronous Packet を再構成して、MPEG2-TS 対応機器にデータを送信する。後者は送信側は機器から受信した Isochronous Packet 全体を IP パケット化する方法である。この方法は送信側のデータの変化を受信側でそのまま再現できる。

本研究では Isochronous Packet 全体を IP パケット化する方法を用いて実装した。この実装により複数の圧縮率に対応した。

本研究により DVTS と同等画質で仕様帯域が少なく、また同等のデータ量で品質の高い映像転送が可能となった。さらに通信環境や用途・目的に応じて、品質の変更可可能な、民生用 AV 機器を利用した映像転送システムを実現した。

7.2 今後の課題

本研究における今後の課題は以下の事項が挙げられる．

- IEEE1394 Isochronous Header を含まない MPEG2-TS の送受信システムの実装
本実装では MPEG2-TS が含まれる IEEE1394 Isochronous Packet ごと送信している．この方法は送信側の IEEE1394 のデータ転送のタイミングを受信側において正確に再現することができるが，本来送るべき MPEG2-TS 以外のデータも送信することになり，使用する帯域が大きくなる．従って，MPEG2-TS のデータのみを送信する方式の実装が必要である．
- パケット損失に対する対策
インターネット上ではパケット損失が起こるが，本実装では パケット損失に対する対策がなされていない．MPEG2-TS はフレーム間圧縮方式であるため，パケット損失が映像に与える影響は大きい．従ってパケット損失を含む通信路誤りを何らかの方法で訂正する必要がある．マルチキャストでの利用を考えるとこの訂正は受信側で行う必要がある．

謝辞

本研究を進めるにあたり，御指導を頂きました，慶應義塾大学環境情報学部教授の村井純博士，徳田英幸博士，同学部助教授の楠本博之博士，中村修博士，同大学環境情報学部専任講師の南政樹氏，重近範行氏に感謝致します。

絶えずご指導とご助言を頂きました独立行政法人通信総合研究所の杉浦一徳氏，慶應義塾大学大学院政策・メディア研究科後期博士課程の小川晃通氏に感謝致します。

また，実装時にご助言を頂きました小川浩司氏，論文執筆時にご助言を頂きました海崎良氏，田原裕市郎氏，小畠元氏，絵を提供してくれた久松剛氏を始めとする STREAM KG の皆様に感謝致します。

参考文献

- [1] Institute of Electrical and Electronics Engineers, Inc. MAC Parameters, Physical Layer, MAUs, and Repeater for 100 Mb/s Operation, Type 100BASE-T. <http://standards.ieee.org/reading/ieee/std/lanman/restricted/802.3u-1995.pdf>, 1998.
- [2] Institute of Electrical and Electronics Engineers, Inc. IEEE Standard-Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications-Physical Layer Parameters and Specification-for 1000 Mb/s Operation over 4-pair of Category 5 Balanced Copper Cabling, Type 1000 BASE-T. <http://standards.ieee.org/reading/ieee/std/lanman/restricted/802.3ab-1999.pdf>, 2000.
- [3] Information technology -Generic coding of moving picture and associated audio information : Systems. *ISO/IEC 13818-1:2000*, 2000.
- [4] Institute of Electrical and Electronics Engineers, Inc. IEEE Std 1394-1995 High Performance Serial Bus. Aug 1996.
- [5] Institute of Electrical and Electronics Engineers, Inc. IEEE Std 1394b-2002, Amendment to IEEE Std 1394-1995. <http://standards.ieee.org/reading/ieee/std/busarch/1394b-2002.pdf>, Dec 2002.
- [6] Microsoft Corporation. PC System Design Guide. <http://www.microsoft.com/hwdev/platform/pcdesign/desguide/default.asp>.
- [7] Compaq, Intel, Microsoft, NEC. Universal Serial Bus Revision 1.1 specification. <http://www.usb.org/developers/docs/usbspec.zip>, Sep 1998.
- [8] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips. Universal Serial Bus Revision 2.0 specification. http://www.usb.org/developers/docs/usb_20.zip, Apr 2000.
- [9] On-The-Go Supplement to the USB 2.0 Specification. http://www.usb.org/developers/onthego/otg1_0.pdf, Dec 2001.

- [10] HD DIGITAL VCR CONFERENCE. Specifications of Consumer-Use Digital VCRs PART1 General Specifications of Consumer-Use Digital VCRs. *Specifications of Consumer-Use Digital VCRs using 6.3mm magnetic tape*, pages 1–61, Dec 1994.
- [11] VICTOR COMPANY OF JAPAN, LIMITED. D-VHS の概要. http://www.vhs-std.com/jpn/dvhs/dvhs_outline.htm, 2001.
- [12] JVC. D-VHS MPEG Transport stream specification Ver1.0-. Jan 2001.
- [13] JVC. D-VHS MPEG Transport stream Service Information specification Ver1.0-. Jan 2001.
- [14] VHS Standard Center JVC. 1394 Ingerface Implementation Guideline for D-VHS Version 1.00. May 2000.
- [15] Sony Marketing (Japan) Inc. MICROMV (マイクロエムヴィ) 方式概要 . <http://www.sony.jp/CorporateCruise/Press/200108/01-0820/micromv.html>, Aug 2001.
- [16] Sony Marketing (Japan) Inc. 『Net MD』規格の概要. <http://www.sony.jp/CorporateCruise/Press/200109/01-0912/netmd.html>, Sep 2001.
- [17] MPEG-2 Generic coding of moving pictures and associated audio information. <http://mpeg.telecomitalialab.com/standards/mpeg-2/mpeg-2.htm>, Oct 2000.
- [18] J. Postel. RFC793 Transmission Control Protocol. <http://www.ietf.org/rfc/rfc0793.txt>, pages 1–85, September 1981.
- [19] J. Postel. RFC768 User Datagram Protocol. <http://www.ietf.org/rfc/rfc0768.txt>, pages 1–3, August 1980.
- [20] Audio-Video Transport Working Group. RFC1889 RTP: A Transport Protocol for Real-Time Applications. <http://www.ietf.org/rfc/rfc1889.txt>, pages 1–75, January 1996.
- [21] Akmichi Ogawa. DVTS(Digital Video Transport Sysytem). <http://www.sfc.wide.ad.jp/DVTS/>.
- [22] 小川 晃通. 協調的輻輳制御を用いた映像配信機構の設計と実装. 1999.
- [23] HD DIGITAL VCR CONFERENCE. Specifications of Consumer-Use Digital VCRs PART2 SD Specifications of Consumer-Use Digital VCRs. *Specifications of Consumer-Use Digital VCRs using 6.3mm magnetic tape*, pages 1–380, Dec 1994.

- [24] K. Kobayashi, A. Ogawa, S. Casner, C. Bormann. RFC3189 RTP Payload Format for DV (IEC 61834) Video. <http://www.ietf.org/rfc/rfc3189.txt>, Jan 2002.
- [25] D. Hoffman, G. Fernando, V. Goyal, M. Civanlard. RFC2250 RTP Payload Format for MPEG1/MPEG2 Video. <http://www.ietf.org/rfc/rfc2250.txt>, Jan 1998.