

卒業論文 2016 年度（平成 28 年度）

映像制作現場における  
高解像度映像 IP 伝送装置の実装と評価

慶應義塾大学 環境情報学部

山中 勇成

徳田・村井・楠本・中村・高汐・バンミーター・植原・三次・中澤・  
武田 合同研究プロジェクト

2017年1月



## 卒業論文 2016 年度（平成 28 年度）

# 映像制作現場における 高解像度映像 IP 伝送装置の実装と評価

## 論文要旨

近年、現行のハイビジョン放送を超える画質である、4K・8K 映像の普及が世界中で加速している。4K 映像の帯域は 2K 映像と比較すると約 4 倍、現行のハイビジョン放送と比較すると約 8 倍にもなり、伝送方法とそのコストが課題である。そのため、映像制作現場では、映像機器と比べて比較的安価なネットワークリソースを活用して伝送する、Video over IP 化が進んでいる。

本研究では、まず映像制作現場の構成を調べ、IP 伝送装置の要件を調査した。要件の 1 つである遅延の許容値を調査するため、映像用のスイッチャーをブラウザ上で再現し、遅延を加えて描画する実験環境を構築した。これを用いて、映像遅延の体感実験を行ったところ、133.33ms の映像遅延まで「遅延を許容できる」と回答した被験者が全体の 9 割であった。したがって、IP 伝送装置の要件として映像遅延は 133.33ms 以内が望ましいことがわかった。

本研究では、この要件を満たす IP 伝送装置としてソフトウェアとハードウェアによる伝送装置の設計、実装を行った。ソフトウェア実装では、4K 映像キャプチャーボードと 10Gbps のネットワークカードを有する汎用マシン上で IP 伝送を行うソフトウェアを実装した。また、ハードウェア実装では、Xilinx の 7-Series FPGA ボードを用いて IP 伝送装置を実装した。

実装したソフトウェアとハードウェアについて、理想的なトラフィックを送信しているか、133.33ms 以内の遅延を達成しているかを計測した。トラフィックを計測した結果、ハードウェア実装では理想的なデータを送信していたが、ソフトウェア実装では処理の問題により理想的なデータを送信できていなかった。また、遅延を計測した結果、ソフトウェアとハードウェアではそれぞれ 199.99ms、33ms 以内の遅延があることがわかった。この結果から、映像制作現場において、ソフトウェア実装では遅延を体感させるが、ハードウェア実装では遅延を感じさせずに、映像制作現場で十分に活用できることがわかった。

## キーワード

4K, IP 伝送, 映像配信システム, FPGA

慶應義塾大学 環境情報学部

山中 勇成

## **Abstract Of Bachelor's Thesis Academic Year 2016**

# **Implementation and Evaluation of Delivery System for High Resolution Video at Video Production Site**

## **Summary**

In recent years, the spread of 4K / 8K images is accelerating all over the world. The bandwidth of 4K video is about 4 times as compared with 2K video, and it is about 8 times as compared with the current high vision broadcasting, and the transmission method and its cost are the issues. For this reason, in the video industry, Video over IP is advancing, utilizing relatively inexpensive network resources compared to video equipment, to transmit.

In this research, we first examined the composition of video production site and investigated the requirements of IP transmission equipment. In order to investigate the tolerance of delay which is one of the requirements, we constructed an experimental environment in which a switcher for video is reproduced on the browser, and drawing is performed with delay added. Using this, we conducted a physical exercise experiment of video delay. As a result, 90Therefore, we found that the video delay is desirable to be within 133.33 ms as a requirement of the IP transmission device.

In this research, we designed and implemented a transmission device by software and hardware as an IP transmission device satisfying this requirement. In software implementation, we implemented software that performs IP transmission on a general purpose machine with 4K video capture board and 10 Gbps network card. In the hardware implementation, an IP transmission device was implemented using Xilinx's 7-Series FPGA board.

We measured whether the ideal traffic is being transmitted or delayed within 133.33 ms for the implemented software and hardware. As a result of measuring traffic, ideal data was transmitted by hardware implementation, but software implementation was not able to transmit ideal data due to processing problems. Also, as a result of delay measurement, it was found that software and hardware had delays of within 199.99 ms and 33 ms respectively. From the results, it was found that at the video production site, software implementation experienced delays, but hardware implementations will not be delayed and can be fully utilized at the video production site.

## **Keywords**

4K, Over IP, Video Streaming, FPGA

Bachelor of Arts in Environmental Information

Keio University

Yusei Yamanaka

## 目 次

<b>第 1 章 序論</b>	<b>1</b>
1.1 本論文の背景 . . . . .	1
1.2 本論文の目的 . . . . .	3
1.3 本論文の構成 . . . . .	3
<b>第 2 章 映像制作現場のアーキテクチャ</b>	<b>5</b>
2.1 映像制作現場の構成 . . . . .	5
2.2 映像機器 . . . . .	6
2.2.1 ビデオカメラ . . . . .	6
2.2.2 ディスプレイ . . . . .	6
2.2.3 スイッチャー . . . . .	7
2.3 帯域の効率化に関する技術 . . . . .	7
2.3.1 インターレース . . . . .	7
2.3.2 色空間と色深度 . . . . .	7
2.4 インターフェース . . . . .	8
2.4.1 VGA . . . . .	8
2.4.2 DisplayPort . . . . .	9
2.4.3 DVI . . . . .	9
2.4.4 HDMI . . . . .	9
2.4.5 SDI . . . . .	11
2.5 帯域 . . . . .	12
2.6 IP 伝送規格 . . . . .	13
2.6.1 SMPTE 2022 . . . . .	13
2.6.2 SMPTE 2110 . . . . .	13
2.6.3 NMI ネットワーク・メディア・インターフェース . . . . .	14
2.6.4 NDI ネットワーク・デジタル・インターフェース . . . . .	14
2.7 遅延 . . . . .	14
2.8 映像制作現場における遅延の許容値の調査 . . . . .	14
2.8.1 実験方法 . . . . .	14
2.8.2 計測結果 . . . . .	17
2.8.3 考察 . . . . .	18
2.9 映像制作現場における IP 伝送装置の要件 . . . . .	18

<b>第3章 システムの設計・実装</b>	<b>21</b>
3.1 ソフトウェアによる実験 . . . . .	21
3.2 ハードウェアによる実装 . . . . .	23
3.2.1 FPGA の回路設計 . . . . .	23
3.3 まとめ . . . . .	30
<b>第4章 評価</b>	<b>31</b>
4.1 トラフィック . . . . .	31
4.1.1 計測手法 . . . . .	31
4.1.2 計測結果 . . . . .	32
4.1.3 考察 . . . . .	35
4.2 遅延 . . . . .	35
4.2.1 計測手法 . . . . .	35
4.2.2 計測結果 . . . . .	36
4.2.3 考察 . . . . .	36
4.3 解像度 . . . . .	38
4.3.1 計測手法 . . . . .	38
4.3.2 計測結果 . . . . .	38
4.3.3 考察 . . . . .	40
4.4 まとめ . . . . .	40
<b>第5章 結論</b>	<b>41</b>
5.1 本研究のまとめ . . . . .	41
5.2 本研究の結論 . . . . .	41
5.3 今後の課題と展望 . . . . .	41
<b>謝辞</b>	<b>43</b>
<b>参考文献</b>	<b>44</b>

## 図 目 次

2.1 現在の映像制作現場における映像機器同士の接続図	5
2.2 将来的な映像制作現場における映像機器同士の接続図	6
2.3 YUV のピクセルあたりの色情報の構造	8
2.4 HDMI ブロックダイアグラム	10
2.5 HDMI 1.4 で定義されている YCbCr 4:4:4 における TMDS マッピング	11
2.6 HDMI 1.4 で定義されている YCbCr 4:2:2 における TMDS マッピング	11
2.7 HDMI 2.0 で定義されている YCbCr 4:2:0 における TMDS マッピング	12
2.8 今回の実験で Web ページ上に再現したマルチビュー映像	15
2.9 実際の中継現場で利用されているマルチビュー映像	16
2.10 遅延計測プログラムのフロー	16
2.11 映像制作現場における遅延の許容結果のグラフ	18
3.1 ソフトウェアによる実装の構成	21
3.2 Blackmagic Design Intensity Pro 4K キャプチャーボード	21
3.3 ソフトウェア全体のブロックダイアグラム	22
3.4 ハードウェアによる実装の構成	23
3.5 TED HDMI 2.0 FMC カード (TB-FMCH-HDMI4K)	23
3.6 FPGA 回路全体のブロックダイアグラム	24
3.7 Independent Clocking FIFO	25
3.8 Video Stream to Ethernet Packet Subsystem Diagram	26
3.9 Ethernet Packet to Video Stream Subsystem Diagram	26
3.10 UDP データの構造	27
3.11 2 つのクロックによるモジュール間でのデータ伝送の波形イメージ	28
3.12 ILA による IP 伝送時の FIFO と Ethernet Subsystem のデータのダンプ	29
3.13 ILA による IP 伝送時の HDMI 入出力のデータのダンプ	29
4.1 ソフトウェア実装における伝送中の受信バイトのグラフ	33
4.2 ソフトウェア実装における伝送中の受信パケットのグラフ	33
4.3 ハードウェア実装における伝送中の受信バイトのグラフ	34
4.4 ハードウェア実装における伝送中の受信パケットのグラフ	34
4.5 遅延の計測手法	35
4.6 ソフトウェア実装による 1 回目の遅延計測のキャプチャー画像	37
4.7 ソフトウェア実装による 4 回目の遅延計測のキャプチャー画像	37

4.8 ハードウェア実装による遅延計測のキャプチャー画像	37
4.9 YCbCr 4:2:0 対応の 4K カメラ FDR-AX1	38
4.10 ハードウェア実装による YCbCr 4:2:0 での出力の様子	39

## 表 目 次

2.1	HDMI 1.4 と 2.0 での 4K(3840x2160) 映像の対応状況	10
2.2	解像度、フレームレート、色空間による HDMI のデータレートの変化	12
2.3	SMPTE 2022 の 7 つの規格の概要	13
2.4	映像制作現場における遅延の許容結果	17
2.5	映像制作現場における IP 伝送装置の要件	19
3.1	ソフトウェアによる実装を検証した PC の構成	21
3.2	10 Gigabit Ethernet Subsystem、及び、Video Processing Subsystem の接続 のために実装したモジュール	25
3.3	Video Processing Subsystem の Axi4-Stream インターフェース	26
3.4	論理合成後のリソース使用状況	30
4.1	ソフトウェア実装による 30FPS における遅延時間の計測結果	36
4.2	実装した伝送装置による YCbCr 4:2:0 の対応状況	38



# 第1章 序論

## 1.1 本論文の背景

近年、現行のハイビジョン放送を超える画質である、4K・8K 映像の普及が世界中で加速している。日本でも東京 2020 オリンピック・パラリンピックに向け、総務省が 4K・8K 映像の普及を後押しをしている。

現在の放送業界では、同軸ケーブルを使用する SDI と呼ばれる伝送規格で映像を伝送する事が一般的である。また、ハイビジョン放送の制作では、1080i と呼ばれる有効走査線数 1080 本のインターレース方式が一般的である。現行のハイビジョンと比較すると、4K 映像の帯域は約 8 倍にもなり、伝送方法とそのコストが課題となっている。

SDI では 4K 映像の伝送を目的として、SMPTE ST-2081[21]、および、SMPTE ST-2082[22] によって、6G-SDI や 12G-SDI が標準化されている。しかし、現行のハイビジョンで使用される HD-SDI と比較すると、高周波による減衰を少なくするために、より太く短い同軸ケーブルを使用しなければならず、現場での扱いづらさが課題である。

多くの放送局のスタジオなどで使用しているカメラでは、カメラ本体と CCU ( カメラコントロールユニット ) を光ファイバーで接続することが主流となっている。しかし、その接続方式は機器やメーカーごとに独自のものであり、光ファイバーのメリットを活用できていない。

そのため近年では、映像機器と比べて比較的安価なネットワークリソースを活用して、映像を IP パケットにして伝送する、Video over IP 化が進んでいる。

SDI による伝送と比較して、IP 伝送には次のようなメリットがある [24]。

- 1 本のケーブルで複数や双方向の映像伝送が可能

同軸ケーブルとは異なり、双方向の映像伝送が 1 本のケーブルで可能である。SDI では、単一の映像伝送を目的としているが、IP 伝送では帯域が許す限り複数の映像や他のソフトウェアのデータなどの付加情報の伝送が可能である。

- 伝送スピードの向上

SDI では、一般的に銅線を使用した同軸ケーブルを使うため、高周波を扱う場合に物理的な限界がある。しかし、IP 伝送で用いられる光ケーブルでは、より高周波を扱うことが可能である。

- コストダウン

ネットワーク機器は、映像制作現場よりも先に 40Gbps や 100Gbps などの帯域に対応し、伝送スピードの高速化が進んでいる。そのため、それらのリソースを活用するこ

とで映像機器に比べて、コストを飛躍的に抑えることが可能である。

また、IP 伝送の規格であるソニーの NMI[12] では、次のようなメリットがある。

- ライブシステムとファイルベースシステムを統合  
リアルタイムなライブ映像の伝送だけではなく、ファイルベースのシステムと統合することが可能である。
- システムの柔軟性  
ネットワーク機器を利用しているため、帯域が許す限り HD から 4K への移行もスムーズに行なうことができる。
- 経路の多重化  
ルーティングシステムに障害が発生した場合、今までケーブルの差し替えやパッチなどで対応することが一般的であったが、IP 網での伝送経路を変更することにより、インフラ設備に対しての可用性を高めることができる。

このように Video over IP 化によるメリットは大きく、映像制作現場において、Video over IP 化が今後進んでいくことは明確である。

## 1.2 本論文の目的

映像の IP 伝送については既に多くの先行研究があり、映像を拠点間などで伝送するための製品なども存在している。しかし、本論文では拠点間の IP 伝送だけにとどまらず、拠点内の設備までを IP 伝送する、Video over IP に着目する。実際の映像制作現場で IP による映像伝送を普及させた際に、現在の拠点間の IP 伝送が抱える課題を洗い出し、拠点内でも快適に IP 伝送を利用することができる要点をまとめ、それが可能であるかについて検証する。

## 1.3 本論文の構成

本論文における以降の構成は次のとおりである。

2 章では、本論文を理解するための前提となる、映像制作現場における構成と映像機器の要素技術についての解説をする。色空間と帯域の関係などにも触れる。同章での解説を基に映像制作現場における IP 伝送装置の要件について述べる。3.1 章では、IP 伝送装置をソフトウェアとハードウェアで設計と実装を行い、その内容について述べる。4 章では、3.1 章で実装したソフトウェアとハードウェアの IP 伝送装置の評価を行い、その結果について考察する。5 章では、本論文のまとめと今後の展望について述べる。



# 第2章 映像制作現場のアーキテクチャ

本章では、映像制作現場を理解するために、映像制作現場の構成を紹介し、技術要素とその仕組みについて解説する。最後に、本研究で IP 伝送装置を実装するにあたり、映像制作現場における IP 伝送の要件をまとめる。

## 2.1 映像制作現場の構成

現在の映像制作現場には、カメラ、スイッチャー、ディスプレイなどをはじめとする多くの映像機器がある。現在の中継現場における、映像機器同士の接続図を図 2.1 に示す。実際の中継現場では、映像の記録を行うためのレコーダー、映像の入出力を切り替えるためのルーターなどあることが多いが、ここでは割愛する。

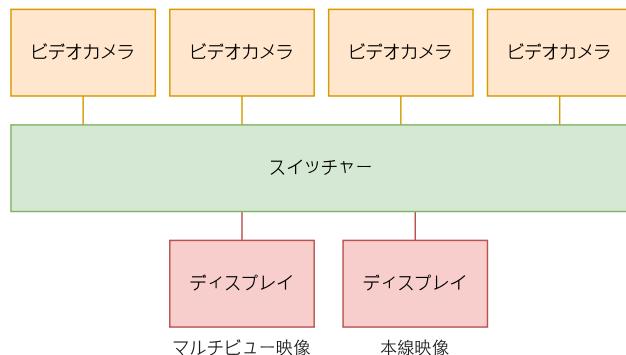


図 2.1: 現在の映像制作現場における映像機器同士の接続図

4台のカメラを1台のスイッチャーに入力し、本線映像が出力される。入力されたソースの映像が複数並んだマルチビュー映像を見てオペレーターが操作することが一般的である。カメラとスイッチャー、スイッチャーとディスプレイは、それぞれ SDI で伝送を行う。

Video over IP 化が進んだ将来、理想的な中継現場における映像機器同士の接続図を図 2.2 に示す。

カメラとスイッチャー、スイッチャーとディスプレイは、それぞれ IP で伝送を行う。カメラからスイッチャーには1本の光ファイバーで接続されているが、スイッチャーからはソース映像、本線映像、マルチビュー映像のために3本の光ファイバーで接続されている。設定解像度の使用する帯域によっては、1本で全ての映像を伝送できる場合もある。

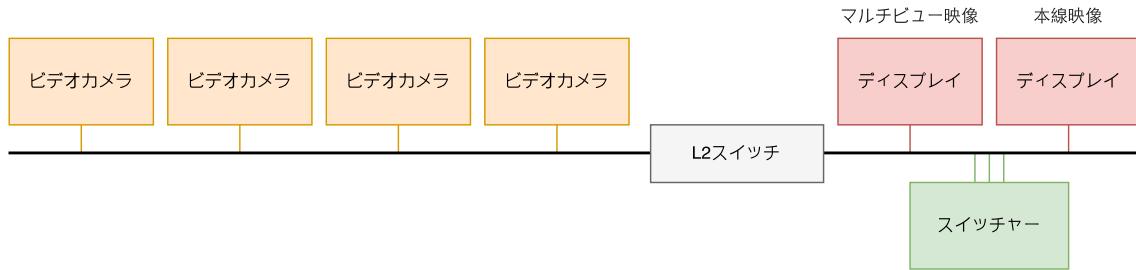


図 2.2: 将来的な映像制作現場における映像機器同士の接続図

## 2.2 映像機器

本節では、図 2.1 と図 2.2 で示した映像制作現場を構成している映像機器について、その解説を述べる。

### 2.2.1 ビデオカメラ

ビデオカメラは、映像の入力機構である。撮影素子であるイメージセンサは、多数の受光素子によって構成されており、それぞれの受光素子は、光エネルギーの明暗に従い電荷を発生する。撮影対象部から反射される光をカメラのレンズを通して、この撮影素子の受光部にあてることで、その明暗を電荷量に光電変換する。変換された電圧値を順次読み出し、電気信号に変換することでアナログ値である光情報をデジタル値に変換する。

ビデオカメラによる出力インターフェースとしては、SDI や HDMI を使用することが一般的である。

### 2.2.2 ディスプレイ

ディスプレイは、映像の出力機構である。ディスプレイには大まかに、アナログディスプレイとデジタルディスプレイに二分できる。

アナログディスプレイでは、CRT、すなわちブラウン管を用いた描画方式である。管面全体を走査線とよぶ固定パターンで走査しつつ、映像信号の輝度成分に従って電子ビームの強さを変調して描画する。このように、画面上の任意の点の明るさを制御することにより画像を作り上げている。

一方、デジタルディスプレイでは、薄い板状の液晶パネルを用いた描画方式である。偏光フィルタから入ってきた光を、電極によりピクセルのカラーごとに電荷をかけることにより、配向膜を光が通り抜け描画する。ブラウン管の走査方式を後継しており、映像の制御信号として、水平同期信号と垂直同期信号が使われている。

### 2.2.3 スイッチャー

スイッチャーは、入力された複数の映像を切り替えて出力する装置である。映像を切り替える際に様々なエフェクトを加えたり、ワイプと呼ばれる Picture in Picture を映像に合成をしたりする機能があるものもある。

放送局などで使用されている業務用のスイッチャーでは、システム全体で同期信号を使用していることが多い。そのため、各入力の信号の同期を取る必要がない。

しかし、ネット配信向けに低価格でコンパクトなスイッチャーでは、ビデオカメラを接続するだけで使えるよう、スイッチャー内部にフレームバッファを備えている。各入力の信号をフレームバッファに蓄積し、スイッチャーのクロックにより映像を合成する。

放送局などでは、できるだけ遅延を短くすることが求められているため、システム全体で同期しているスイッチャーのほうが有利である。

## 2.3 帯域の効率化に関する技術

本節では、映像データを高品質のまま効率の良い帯域で映像伝送をする技術について述べる。テレビジョン放送など、帯域の限られた映像伝送の環境では必須の技術であり、様々な技術がある。最近では、インターネットによる映像伝送も普及しており、帯域の制限はシビアでなくなっている。

なお、フレーム間圧縮やフレーム内圧縮の技術については、ここでは触れない。

### 2.3.1 インターレース

画像伝送において、データレートを増やすずに描画回数を増やす技術である。この方式の特徴は、「人間の視野は動くものの細部を捉えられない」という性質に基づいている。飛び越し走査とも呼ばれ、奇数番目の走査線を先に送り、偶数番目の走査線をその後に送る。

デジタル化が進んでいる現在でも、走査線を 1 ラインに割り当て、データレートを減らす際の手段として利用されている。また、インターレースではない、画像をプログレッシブ呼び、インターレースからプログレッシブに変換する処理を、ディインターレースと呼ぶ。

主な使用例として、日本のテレビジョン放送では、アナログテレビ放送、デジタルテレビ放送のどちらでも使われている。幸いなことに、4K・8K 映像ではインターレース方式は使われることはない。

### 2.3.2 色空間と色深度

一般的に液晶ディスプレイでは、1 ピクセルを赤、緑、青、すなわち RGB の 3 つの色信号で表現する。多くの PC やゲーム機の出力では RGB の色空間が使われ、RGB それぞれ

8bit、1ピクセルあたり24bitで表現する。1ピクセルあたりを表現するビット数を色深度といい、色解像度、色分解能とも言われる。24bitの色深度では、16,777,216色を表現することができる。

一方、ビデオカメラでは、輝度信号Yと2つの色差信号を使って表現される色空間であるYUVが使われることが多い。この方式の特徴は、「人間の目は明るさの変化には敏感だが、色の変化には鈍感である」という性質に基づいて、色度信号の情報量を減らすことができるという点にある。

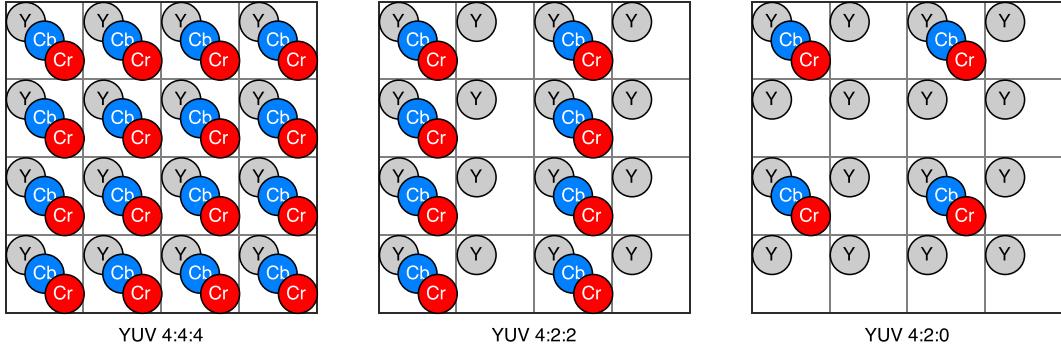


図 2.3: YUV のピクセルあたりの色情報の構造

YUV 4:4:4では、輝度信号、色差信号共に1ピクセル毎である。YUV 4:2:2では、輝度信号は1ピクセル毎、色差信号は2ピクセル毎であり、同じ色深度のYUV 4:4:4と比べ、帯域はおよそ $2/3$ となる。YUV 4:2:0では、輝度信号は1ピクセル毎、色差信号は4ピクセル毎であり、同じ色深度のYUV 4:2:2と比べ、帯域はおよそ $3/4$ となり、同じ色深度のYUV 4:4:4と比べ、帯域はおよそ $1/2$ となる。

なお、図 2.3 で示した、UV成分である Cb、Cr の色のサンプリング方法は、伝送方式の規格によって定められている。

## 2.4 インターフェース

本節では、映像機器で使われているインターフェースについてまとめ、それぞれの特徴について述べる。

### 2.4.1 VGA

VGA (Video Graphics Array) は、IBMが発表したアナログ映像信号の伝送規格、または、同社が開発したVGA表示回路用のチップのことを指す。DE-15コネクタを使用し、赤、緑、青、垂直同期、水平同期の5つのアナログ信号で映像を伝送することができる。DDC (VESA Display Data Channel) 信号を使用することで、接続機器の対応する解像度を送信することができ、最近では1080pの映像を伝送する機器も多い。PCでの映像出力方式とし

て普及したが、アナログ信号であることや、音声伝送の手段が別途必要となるため、HDMI や DisplayPort などのインターフェースに移行が進んでいる。

#### 2.4.2 DisplayPort

DisplayPort は、VESA<sup>1</sup> によって標準化された映像伝送規格であり、主に超解像度向けのインターフェースとして普及している。DisplayPort 1.3 からは、32.4 Gbps のデータレートに対応し、8K 映像の伝送にも対応している。

#### 2.4.3 DVI

DVI ( Digital Visual Interface ) は、VESA<sup>1</sup> によって標準化された デジタル映像信号の伝送規格である。物理層として、Silicon Image が開発した TMDS ( Transition Minimized Differential Signaling )を使用している。TMDS は、データの 3 チャネルとクロックの 1 チャネルを備えた 4 つのツイストペアケーブルで構成され、主に高速シリアル通信で使用されている。TMDS では、データの 8b/10b 符号化が行われ、データレートは 20% のロスとなるが、DC 成分の偏りを押さえ、プランギング区間などでも I/O の遷移を増やしてデータの境界検出を用意している。

#### 2.4.4 HDMI

HDMI ( High-Definition Multimedia Interface ) は、映像、音声をデジタル信号で伝送する通信インターフェースの規格である。DVI を基に、音声伝送機能や著作権保護機能を加えたものであり、物理層は DVI と同じ TMDS を使用している。

HDMI のシステム構成は大きく分けて、映像を送る機器 ( Source ) 映像を受け取る機器 ( Sink ) ケーブルの 3 つに分類することができる。

HDMI 2.0 では、帯域を 18Gbps に拡大し、4K@60p に対応している。また、CES 2017 に合わせ、HDMI 2.1 が発表され、帯域を 48Gbps に拡大し、8K@60p に対応した。

HDMI 1.4[14] では、RGB、YCbCr 4:4:4、YCbCr 4:2:2 の色空間がサポートされている。HDMI 2.0[15] では、4K 解像度向けに YCbCr 4:2:0 の色空間がサポートされた。YCbCr 4:2:0 によるピクセルエンコーディングの規格では、YCbCr 4:2:2 と比べ、1/2 のデータレートで転送することが可能となった。これにより、一部の機器では 4K 解像度への対応をソフトウェアだけで行なうことが可能である。

YCbCr 4:4:4、YCbCr 4:2:2 の TMDS データのマッピングを、図 2.5 と図 2.6 に示す。2.3.2 項では、同じ色深度の場合 YUV 4:2:2 は YUV 4:4:4 と比べ帯域が 2/3 になると述べたが、HDMI 1.4 で定義されている YCbCr 4:2:2 では、1 ピクセルあたりの色深度は変わらず、

---

<sup>1</sup> Video Electronics Standards Association ビデオ周辺機器に関する業界標準化団体

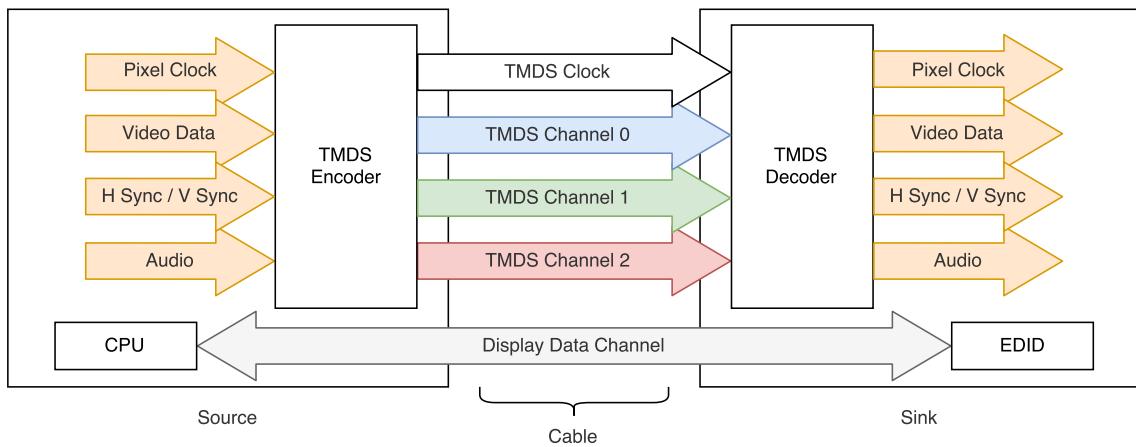


図 2.4: HDMI ブロックダイアグラム

表 2.1: HDMI 1.4 と 2.0 での 4K(3840x2160) 映像の対応状況

フレームレート	ピクセルあたりの色深度	HDMI 1.4	HDMI 2.0
30Hz	24bit	対応	対応
	30bit	対応	対応
	36bit	対応	対応
	48bit	非対応	対応
60Hz	24bit	非対応	対応
	30bit	非対応	対応
	36bit	非対応	対応
	48bit	非対応	非対応

Y および CbCr のサンプリング解像度が 8bit から 12bit になる。そのため、HDMI では色空間の YCbCr 4:4:4、 YCbCr 4:2:2 のどちらであっても帯域には影響しない。

TMDS 0 [7:0]	Cb <sub>0</sub>	Cb <sub>1</sub>	Cb <sub>2</sub>	Cb <sub>3</sub>	Cb <sub>4</sub>	...
TMDS 1 [7:0]	Y <sub>0</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	...
TMDS 2 [7:0]	Cr <sub>0</sub>	Cr <sub>1</sub>	Cr <sub>2</sub>	Cr <sub>3</sub>	Cr <sub>4</sub>	...

図 2.5: HDMI 1.4 で定義されている YCbCr 4:4:4 における TMDS マッピング

TMDS 0 [3:0]	Y <sub>0</sub> [3:0]	Y <sub>1</sub> [3:0]	Y <sub>2</sub> [3:0]	Y <sub>3</sub> [3:0]	Y <sub>4</sub> [3:0]	...
TMDS 0 [7:4]	Cb <sub>0</sub> [3:0]	Cr <sub>0</sub> [3:0]	Cb <sub>2</sub> [3:0]	Cr <sub>2</sub> [3:0]	Cb <sub>4</sub> [3:0]	...
TMDS 1 [7:0]	Y <sub>0</sub> [11:4]	Y <sub>1</sub> [11:4]	Y <sub>2</sub> [11:4]	Y <sub>3</sub> [11:4]	Y <sub>4</sub> [11:4]	...
TMDS 2 [7:0]	Cb <sub>0</sub> [11:4]	Cr <sub>0</sub> [11:4]	Cb <sub>2</sub> [11:4]	Cr <sub>2</sub> [11:4]	Cb <sub>4</sub> [11:4]	...

図 2.6: HDMI 1.4 で定義されている YCbCr 4:2:2 における TMDS マッピング

YCbCr 4:2:0 の TMDS データのマッピングを、図 2.7 に示す。HDMI では、24bit の他に、30bit、36bit、48bit の色深度に対応しているが、YCbCr 4:2:0 では、24bit のみの対応である。

#### 2.4.5 SDI

SDI ( Serial Digital Interface ) は、SMPTE<sup>2</sup> によって標準化された映像伝送規格であり、主に業務機器向けの規格である。

同軸ケーブルを使用しているため、HDMI と比べて距離に対する減衰が少なく、HD-SDI では、およそ 100m 遠方に伝送することができる。BNC 端子を使用することが一般的であり、抜け落ち防止のためのロック機能があるため、放送局や中継現場で使われる。

解像度や帯域に応じて、SD-SDI、HD-SDI、3G-SDI、6G-SDI、12G-SDI など複数の規格が定められている。また、4K・8K 映像を伝送するために、HD-SDI や 3G-SDI を 2 本 1 組や 4 本 1 組で使用して伝送する規格も定められている。

<sup>2</sup>米国映画テレビ技術者協会

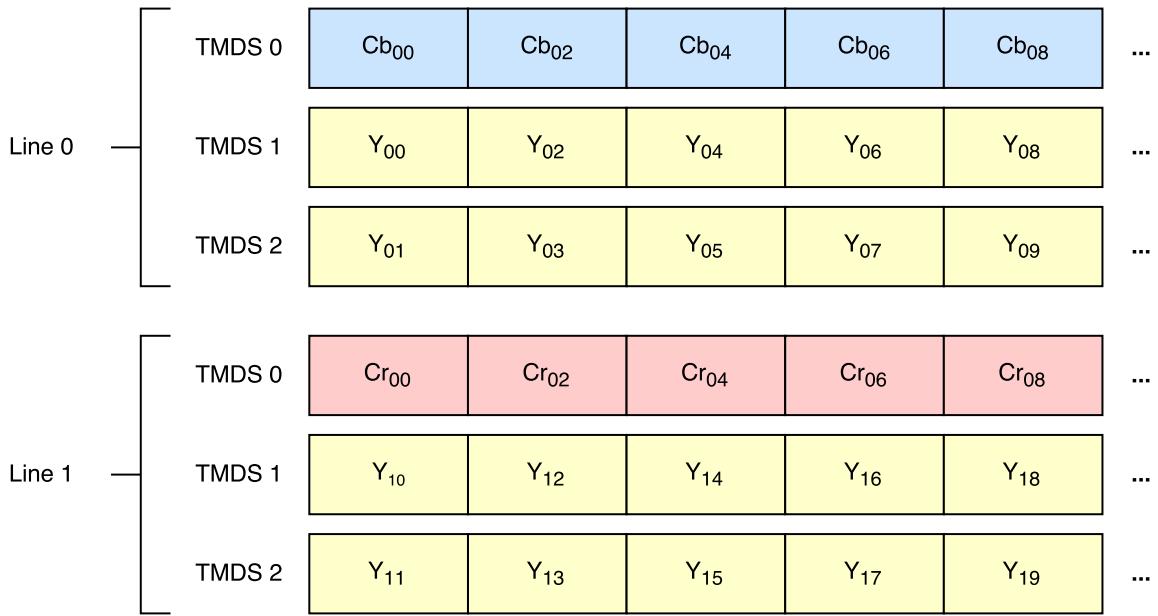


図 2.7: HDMI 2.0 で定義されている YCbCr 4:2:0 における TMDS マッピング

## 2.5 帯域

帯域は解像度の他にも、インターレース、色空間、色深度により変化する。また、伝送するインターフェースの規格によっても、物理層での扱いにより若干の違いがある。ここでは、HDMI で色深度を 8bit とした場合の解像度、フレームレート、色空間別に見たピクセルクロック、データレートを表 2.2 に示す。

表 2.2: 解像度、フレームレート、色空間による HDMI のデータレートの変化

解像度	フレームレート	色空間	ピクセルクロック	データレート
3840x2160	60p	RGB	594MHz	17.82 Gbps
3840x2160	60p	YUV422	594MHz	17.82 Gbps
3840x2160	60p	YUV420	297MHz	8.91 Gbps
3840x2160	30p	RGB	297MHz	8.91 Gbps
3840x2160	30p	YUV422	297MHz	8.91 Gbps
1920x1080	60p	RGB	148.5MHz	4.455 Gbps
1920x1080	60p	YUV422	148.5MHz	4.455 Gbps
1920x1080	60i	RGB	74.25MHz	2.2275 Gbps
1920x1080	60i	YUV422	74.25MHz	2.2275 Gbps

同期区間を含めた垂直ピクセルを  $p_w$ 、水平ピクセルを  $p_h$ 、ピクセルあたりのビット数を  $b$ 、フレームレートを  $f$  としたとき、HDMI のデータレート  $r$  は次のようにして求めること

ができる。HDMI の物理層である TMDS では、8b/10b 変換が行われるため、データレートとしては 1.25 倍となる。

$$r = 1.25bfp_wph$$

## 2.6 IP 伝送規格

Video over IP における映像の IP 伝送規格は、SMPTE 2022 と NMI が主流となっている [23]。しかし、その他にも多くの規格が提唱され、市場ではどの規格で統一されるかが静観されている。

IP 伝送規格は、SDI などの標準化を行っている SMPTE や、映像制作現場などの機器を制作している会社が製品とともに規格化を行う事が多い。この節では、抜粋して幾つかの IP 伝送規格について解説する。

### 2.6.1 SMPTE 2022

SMPTE 2022[18] は、SMPTE が提唱、標準化した IP 伝送規格であり、表 2.3 に示す 7 つの規格に分かれている。

表 2.3: SMPTE 2022 の 7 つの規格の概要

規格	概要
SMPTE 2022-1	IP 伝送でのリアルタイムビデオ/オーディオ転送の FEC 訂正
SMPTE 2022-2	IP 伝送での固定ビットレート MPEG-2 TS の単方向転送
SMPTE 2022-3	IP 伝送での可変ビットレート MPEG-2 TS の単方向転送
SMPTE 2022-4	IP 伝送での非ピース単位の可変ビットレート MPEG-2 ストリームの単方向転送
SMPTE 2022-5	IP 伝送での高ビットレートメディア信号の伝送のための前方誤り訂正
SMPTE 2022-6	IP 伝送でのネットワークを介した高ビットレートメディア信号の伝送
SMPTE 2022-7	IP データグラムのシームレスな保護スイッチング

SMPTE 2022-1/2/3/4 では、MPEG2 圧縮をベースとした IP 伝送について規格化され、SMPTE 2022-5/6 では、非圧縮であり SDI のペイロードを基とした IP 伝送について規格化されている。

### 2.6.2 SMPTE 2110

SMPTE 2110[19] は、SMPTE が制定中の規格であり、VSF ( Video Services Forum ) に提出された TR03、TR04 の内容を取り込んでいる。

SMPTE 2022-5/6 では SDI のペイロードを基としているため、IP パケットにする際には SDI をカプセル化している。そのため、映像と音声データを IP レイヤーから識別することができず、制御に利用しにくいなどの問題がある。この問題を回避するため、SMPTE 2110 では、ビデオデータの伝送には RFC 4175[13] の RTP、音声データの伝送には AES 67 を使用するなど、より効果的な IP 伝送規格になるよう設計されている。

### 2.6.3 NMI ネットワーク・メディア・インターフェース

NMI[12] は、ソニービジネスソリューションが提唱、規格化した IP 伝送規格である。

非圧縮ではなく、低遅延高画質のコーデックであり、Visually Lossless な LLVC[20] によって圧縮されている。また、機器間の同期にはナノ秒レベルの高精度同期が行える、SMPTE ST2059 を使用している。

### 2.6.4 NDI ネットワーク・デジタル・インターフェース

NDI[17] は、NewTek が提唱、開発したオープンな IP 伝送規格である。

多くの IP 伝送規格は商用向けであり、詳細な仕様はオープンになっていないが、NDI では SDK やプラグインなどを公開し、ユーザーを集めている。同社では IP ワークフローとして、NDI を利用したスイッチャーや入出力システムなどを提供している。

## 2.7 遅延

遅延は、ケーブルなどの物理的なものでも発生するが、人の目に気になるほどの遅延は映像機器などの処理で発生することが多い。

中継現場で拠点間の映像の IP 伝送であれば、ある程度の遅延を許容することができる。屋外での中継で、外にいるリポーターと局内にいるキャスターとの音声に遅延があり、やり取りに間がある光景を見ることは少なくない。しかし、拠点内の映像の IP 伝送では、映像と音声が同期している必要があり、遅延はシビアな問題となる。

## 2.8 映像制作現場における遅延の許容値の調査

前節で述べた遅延について、本実装での要件をまとめたため、映像制作現場においてオペレーターが許容できる遅延の範囲を調査する実験を行った。

### 2.8.1 実験方法

ブラウザ上で、キーボードの入力を利用した擬似的なスイッチャーの操作を行い、マルチビュー映像の出力を行うプログラム [9] を開発した。音声は出力されない。この実験では、

図 2.2において、スイッチャーとマルチビュー映像を表示するためのディスプレイ間での遅延が許容できるかについて調査したことと同義である。プログラムで出力されるマルチビュー映像を図 2.8 に示す。これは、実際の中継現場で利用されているマルチビュー映像とほぼ同じである。



図 2.8: 今回の実験で Web ページ上に再現したマルチビュー映像

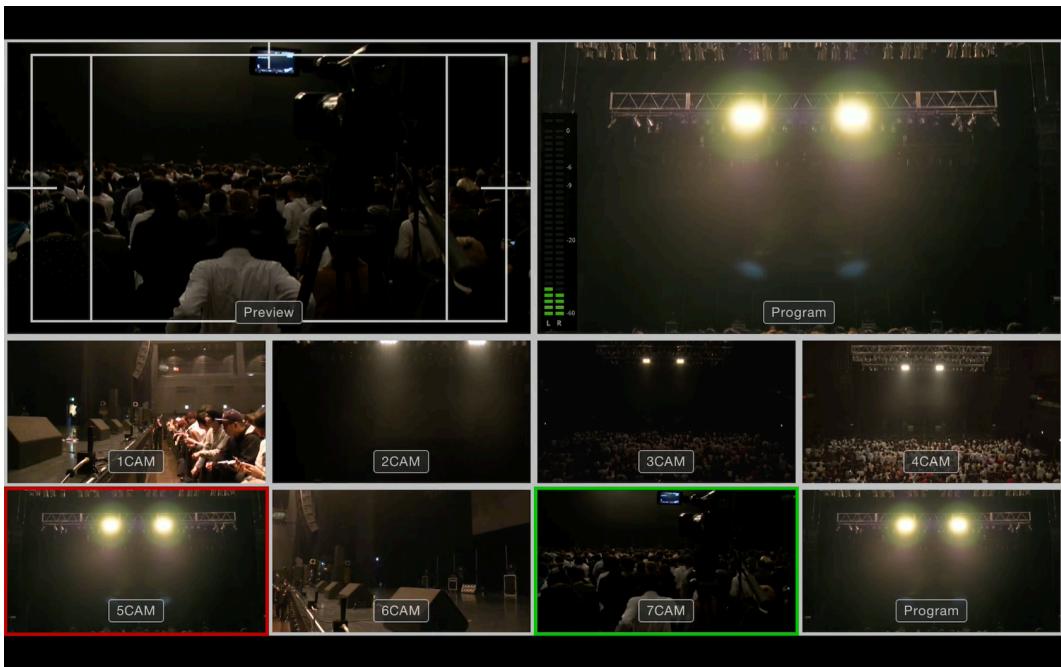


図 2.9: 実際の中継現場で利用されているマルチビュー映像

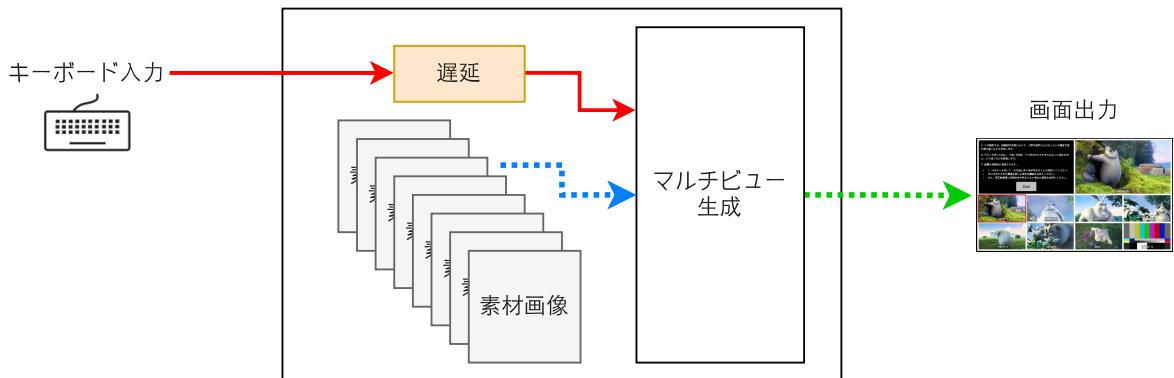


図 2.10: 遅延計測プログラムのフロー

プログラムは HTML と JavaScript を用いて作成した。遅延計測プログラムのフローを図 2.10 に示す。プログラムでは、キーボードからの入力に決められた遅延を与え、マルチビューの操作を行う。マルチビュー生成では、各入力との遅延処理されたキーボードからの入力に基づいて出力する。各入力の素材としては、Creative Commons で提供されている Big Buck Bunny[2] の動画から、いくつかのシーンをキャプチャーした静止画を使用した。

実験では、0ms から 499.99ms までを、30FPS の 1 フレームに相当する 33.33ms 間隔で分けた 16 段階のステップにわけ、各段階で入力から表示までの遅延を与える。各段階で与える遅延については、心理的な判断を避けるため実験終了後まで表示を行わない。また、各段階で与える遅延は、実験ごとにランダムになっている。各段階では、被験者からの入力を 15

秒間受け付ける。各段階の終了後、被験者に「遅延を許容できる」か「遅延を許容できない」かを問う。

実験では、ブラウザのレンダリングによる遅延も発生するため、キーボードの入力からブラウザのレンダリングによる遅延も計測した。具体的には、キーボードの入力時間から、JavaScript における次の `window.requestAnimationFrame` イベントが呼ばれた時間までの差分である。

被験者の対象は、映像制作現場に関わったことがある人で、マルチビュー映像に理解がある人と限定した。被験者は 38 人である。

### 2.8.2 計測結果

全被験者からの総入力回数は、10,645 回であった。キー入力からブラウザのレンダリングによる遅延の平均は、8.6889ms であった。ブラウザのレンダリングによる遅延は、60FPS における 1 フレーム未満であるため、ここでは無視できるものとした。

それぞれの遅延時間における「遅延を許容できる」と回答した人数を表 2.4 に示す。

表 2.4: 映像制作現場における遅延の許容結果

遅延時間	「遅延を許容できる」と回答した人数
0 ms	38 人中 37 人
33.33 ms	38 人中 38 人
66.66 ms	38 人中 36 人
99.99 ms	38 人中 35 人
133.33 ms	38 人中 35 人
166.66 ms	38 人中 31 人
199.99 ms	38 人中 28 人
233.33 ms	38 人中 26 人
266.66 ms	38 人中 18 人
299.99 ms	38 人中 13 人
333.33 ms	38 人中 14 人
366.66 ms	38 人中 15 人
399.99 ms	38 人中 15 人
433.33 ms	38 人中 14 人
466.66 ms	38 人中 13 人
499.99 ms	38 人中 10 人

### 2.8.3 考察

映像制作現場における遅延の許容結果のグラフを図 2.11 に示す。縦軸は「遅延を許容できる」と回答した人数であり、横軸は各段階で与えた遅延時間である。

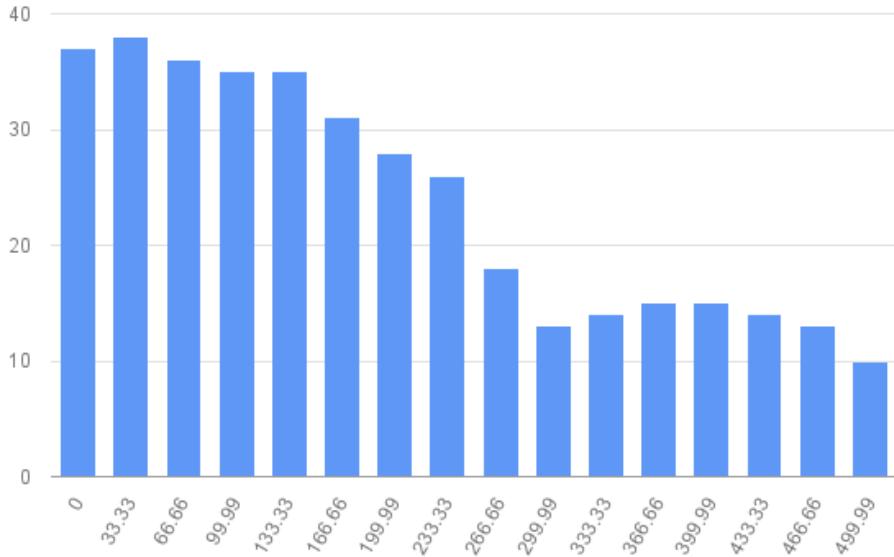


図 2.11: 映像制作現場における遅延の許容結果のグラフ

133.33msまでの遅延では、被験者の9割以上が「遅延を許容できる」と回答しているが、166.66msの遅延では、8割を下回った。また、233.33msから266.66msの遅延になると、被験者の過半数が「遅延を許容できない」と回答した。その他、グラフから読み取れる点として、299.99msから366.66msで人数が多少増加している。この原因としては、前述の通り各段階で与える遅延の順番はランダムであるため、前段階の遅延よりも短かった場合に体感的に「遅延を許容できる」と回答してしまった人がいると考える。

遅くとも133.33msの遅延であれば、9割以上が遅延を許容できるため、過酷な映像制作現場に遅延が許容できると考えられる。133.33msは、30FPSで4フレーム、60FPSで8フレームである。

## 2.9 映像制作現場におけるIP伝送装置の要件

本章では、主に映像制作現場の構成を解説し、技術要素を解説した。本節では、本論文で実装をするIP伝送装置の要件をまとめた。

映像制作現場では、インターフェースとしてSDIが用いられることが多い。しかし、SDIをインターフェースとする開発環境を整えるためには、コストが問題となる。そのため、本実装ではコストを押さえつつ、SDIの次に映像制作現場で使われているHDMIをインターフェースとする。

解像度とフレームレートについては、計算上、10Gbps の帯域で 4K 30P の映像だけでなく、色空間と色深度を YCbCr 4:2:0 とすることで 4K 60P の映像も伝送することが可能であった。近年では、より高い解像度とフレームレートが求められるため、本実装では 4K 30P の他に、YCbCr 4:2:0 方式で 4K 60P の映像が伝送できることを要件とする。

IP 伝送規格については、SMPTE 2022 と NMI が主流であるが、今回は評価と実装のしやすさを考慮し、独自方式のプロトコルを用いる。

遅延については、前節の実験結果より映像制作現場では 133.33ms 以内の遅延であることが望まれる。

表 2.5: 映像制作現場における IP 伝送装置の要件

インターフェース	HDMI
対応解像度	4K 3840x2160 30P
	4K 3840x2160 60P (但し YCbCr 4:2:0 方式)
プロトコル	独自規格
遅延時間	133.33ms 以内
トラフィック	10Gbps 以内



# 第3章 システムの設計・実装

本章では、2章で述べた映像制作現場におけるIP伝送装置の要件について評価するため、4K映像を非圧縮IP伝送をするシステムをソフトウェアとハードウェアで設計と実装を行い、その内容について述べる。

## 3.1 ソフトウェアによる実験

ソフトウェアによる実装では、汎用的なPCに10Gbpsに対応したNIC、4K対応キャプチャーボードBlackmagic Design Intensity Pro 4Kを用いて行った。本実装の概要を、図3.1に示す。

表3.1: ソフトウェアによる実装を検証したPCの構成

OS	Ubuntu 14.04 Desktop
CPU	Intel Core i7-4770 @ 3.40GHz
RAM	8GB

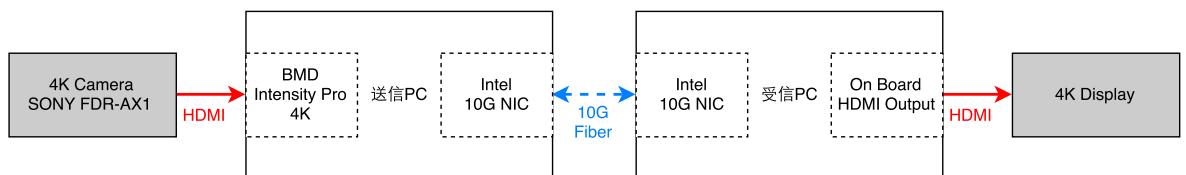


図3.1: ソフトウェアによる実装の構成



図3.2: Blackmagic Design Intensity Pro 4K キャプチャーボード

送信プログラムでは、キャプチャーボードのデータを Blackmagic DeckLink SDK[16] を用いて取得し、IP 経由で伝送するプログラムを作成した。受信プログラムでは、IP 経由で受信したデータを Linux 汎用的なメディアプレーヤーである mplayer で再生するプログラムを作成した [10]。今回はダークファイバー環境での想定であり、順序制御、再送制御の実装を省くため、TCP で実装を行った。

ソフトウェア全体のブロックダイアグラムを図 3.3 に示す。

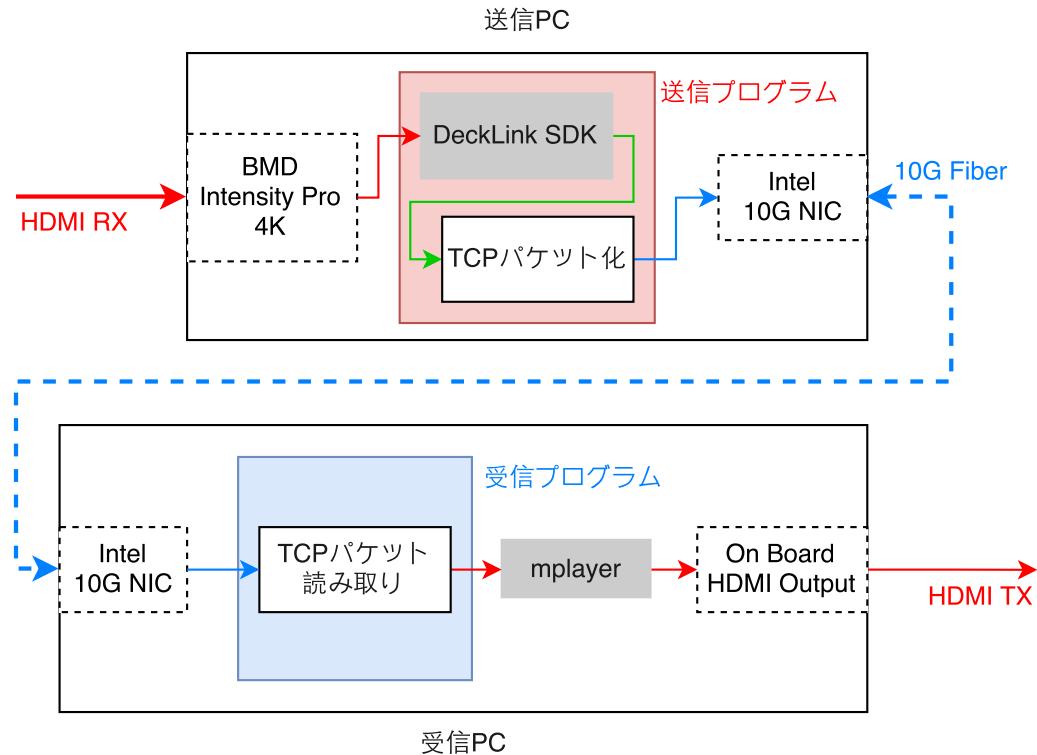


図 3.3: ソフトウェア全体のブロックダイアグラム

送信 PC では映像データを DeckLink SDK 経由で処理する。キャプチャーボードと SDK の仕様によりデータはフレーム単位でしか処理することができない。この時点で 1 フレームの遅延の発生が懸念される。受信プログラムでは、映像データを TCP パケット化して送信する。送信するためのネットワークインターフェースの制御はカーネルに依存する。

受信 PC では、ネットワークインターフェースから受信した TCP パケットを、受信プログラムで処理する。受信した映像データを Linux の汎用プレーヤーである mplayer に受け渡し、画面上で映像が再生される。VRAM の操作やビデオカードによる出力を行っていた。この時点でも OS のタイミングで描画されるため、1 フレーム程の遅延の発生が懸念される。

### 3.2 ハードウェアによる実装

IP 伝送装置は、Xilinx KC705 評価ボード [8]、HDMI インターフェースカードである TED HDMI 2.0 FMC カード [7] を使用した。本実装の構成を図 3.4 に示す。

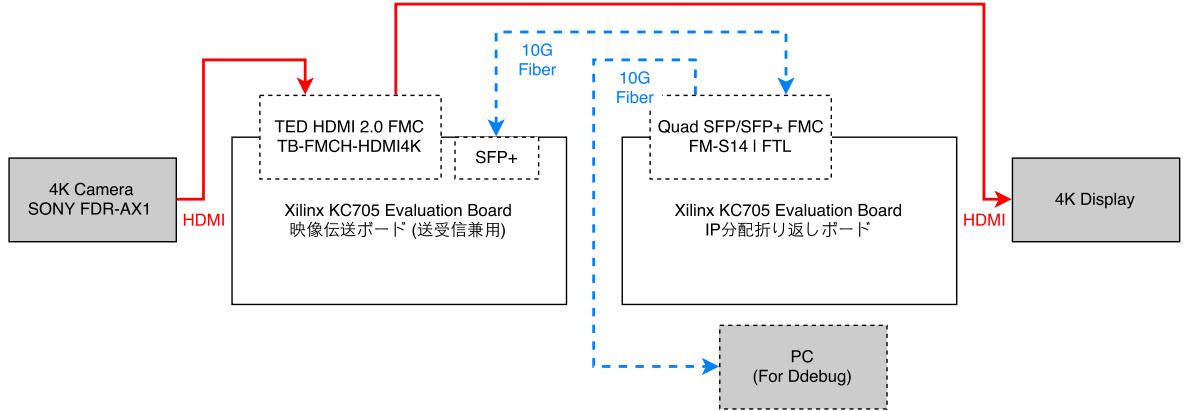


図 3.4: ハードウェアによる実装の構成



図 3.5: TED HDMI 2.0 FMC カード (TB-FMCH-HDMI4K)

今回の構成では、IP 伝送装置とは別に 1 台の Xilinx KC705 評価ボード、Quad SFP/SFP+ カードを用いて、送信した IP パケットを折り返しする装置を使用した。IP パケットを折り返しする装置について、IP 伝送装置と直接の関係はないため、ここでの解説は割愛する。

本実装では、論理合成ツールとして Xilinx Vivado 2016.2 を使用した。また、開発言語として Verilog HDL を使用した。

#### 3.2.1 FPGA の回路設計

本実装は、Xilinx が提供している Kintex-7 シリーズ向けの HDMI 2.0 のリファレンス実装である xapp1287[6] をベースとしている。Xilinx の提供する IP である 10 Gigabit Ethernet Subsystem[1]、Video PHY Controller[11]、HDMI 1.4/2.0 Transmitter Subsystem[5]、及

び、HDMI 1.4/2.0 Receiver Subsystem[4]、FIFO Generator[3] が使用されている。本研究のために新たに実装をした箇所は、これらの IP に対してデータを受け渡しするモジュールとそのモジュールを含んだ Ethlogic Subsystem である。FPGA の回路全体のブロックダイアグラムを図 3.6 に示す。なお、IP 伝送装置に関するモジュールのみを掲載している。

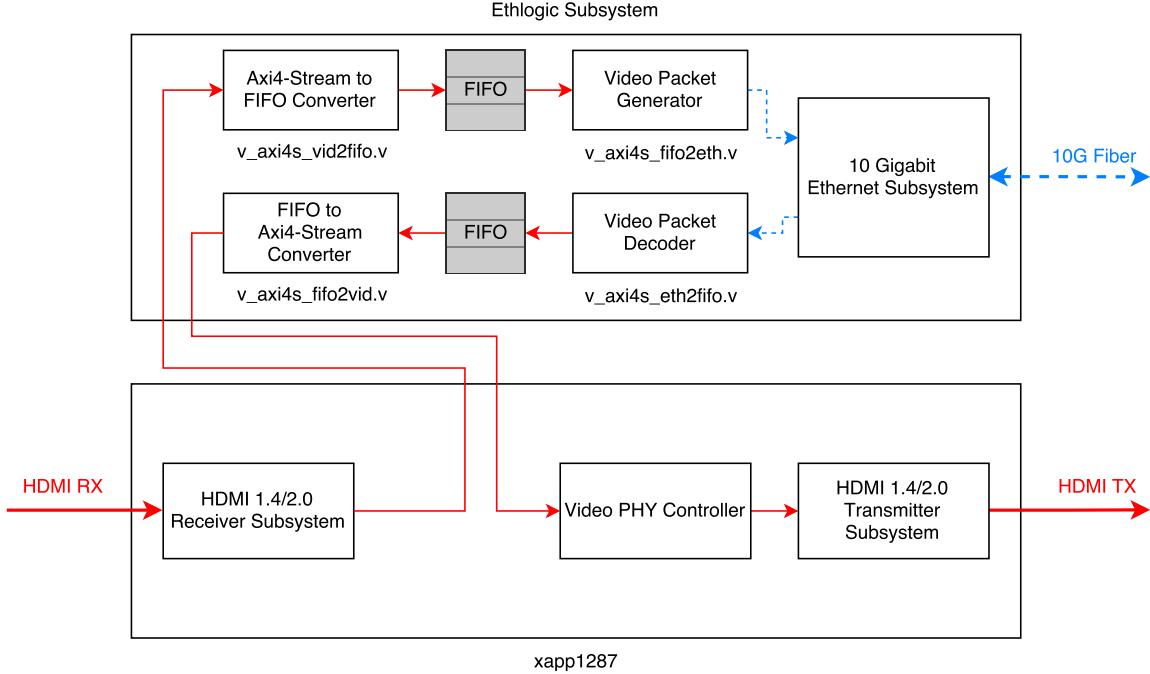


図 3.6: FPGA 回路全体のブロックダイアグラム

受信側と送信側で、10 Gigabit Ethernet Subsystem と Video Processing Subsystem 間の受け渡しを行うため、表 3.2 に示す 4 つのモジュールを実装した。

10 Gigabit Ethernet Subsystem の基準クロックは 64bit 幅の設定で 156.25MHz となり、Video Processing Subsystem の基準クロックは 300MHz となる。互いの基準クロックが異なるため、データをそのまま受け渡しすることはできない。

この問題を解決するため、読み書きで独立したクロックに対応した Independent Clocking FIFO を FIFO Generator で作成する。今回の IP 伝送装置の送信側で用いた、FIFO モジュールを図 3.7 に示す。

読み書きで独立したクロックの他に、入出力のデータ幅が異なっており、入力のデータ幅は 35bit、出力のデータ幅は倍の 70bit となっている。理由は後述する。Video Processing Subsystem は 300MHz で 35bit のデータを書き込み、10 Gigabit Ethernet Subsystem は 156.25MHz のデータを読み込む。10 Gigabit Ethernet Subsystem のクロックが早いため、FIFO がフル状態になることはない。また、almost\_empty フラグを使用しており、empty になる 1 クロック前に知ることが可能である。

表 3.2: 10 Gigabit Ethernet Subsystem、及び、Video Processing Subsystem の接続のため  
に実装したモジュール

Name	Description
v_axi4s_eth2fifo.v	Ethernet Subsystem のクロックで、Ethernet Subsystem から送られてきた映像データを FIFO に書き込むモジュール
v_axi4s_fifo2eth.v	Ethernet Subsystem のクロックで、FIFO から読み込んだ映像データを Ethernet Subsystem に送るモジュール
v_axi4s_vid2fifo.v	Video Processing Subsystem のクロックで、Video Processing Subsystem から送られてきた映像データを FIFO に書き込むモジュール
v_axi4s_fifo2vid.v	Video Processing Subsystem のクロックで、FIFO から読み込んだ映像データを Video Processing Subsystem に送るモジュール



図 3.7: Independent Clocking FIFO

これまでに解説したモジュールの組み合わせについて、送信側の接続を図 3.8 に示す。

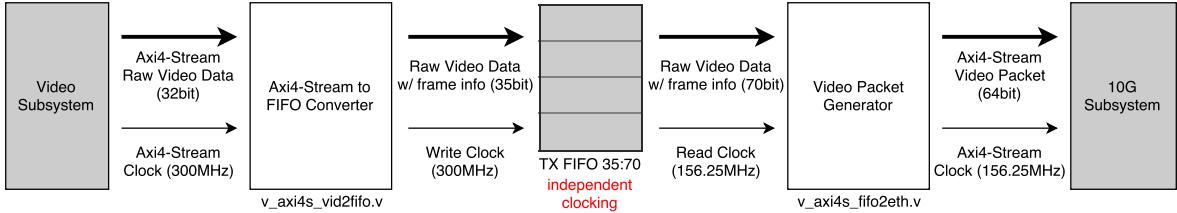


図 3.8: Video Stream to Ethernet Packet Subsystem Diagram

同様に受信側の接続を図 3.9 に示す。

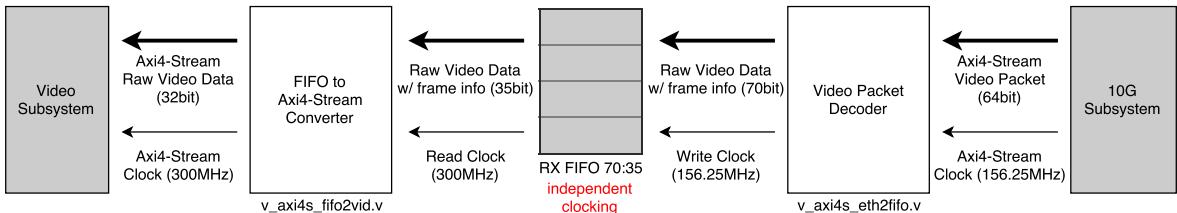


図 3.9: Ethernet Packet to Video Stream Subsystem Diagram

Video Processing Subsystem が output する Axi4-Stream の tdata は映像データを表し、有効データ幅は 32bit である。また、表 3.3 に示すとおり、tlast がラインの終了、tuser がフレームの開始を表す。FIFO に映像データだけを書き込んだ場合、ラインの終了、フレームの開始のタイミングが失われることとなる。この問題を解決するため、FIFO には Axi4-Stream の tdata の他に、tlast、tuser、tvalid も書き込む。

表 3.3: Video Processing Subsystem の Axi4-Stream インターフェース

Name	Width	Description
tdata	$3 \times \text{BPC}^3 \times \text{PPC}^4$	Data
tlast	1	End of line
tready	1	Ready
tuser	1	Start of frame
tvalid	1	Valid

図 3.10 に、本実装で用いた UDP データの構造を示す。UDP データの映像データより前のヘッダー区間は 6bytes となっている。これは、FPGA 内部で Ethernet パケットを構築していくと、データの先頭 6bytes が丁度 1 クロックで送る 64bit の区切り目となるためであ

<sup>3</sup>Max Bits Per Component

<sup>4</sup>Pixels Per Clock

り、FPGAで処理する際に効率が良い。UDPのパケットは FIFO にデータがある間生成され続けるため、IP パケット上の長さは 0 としている。映像データによってはジャンボフレームとなる場合もある。

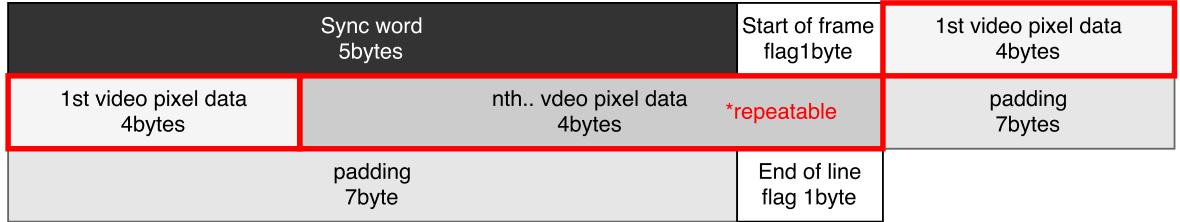


図 3.10: UDP データの構造

各モジュールで映像データがどのように扱われるかの波形イメージを、図 3.11 に示す。HDMI 1.4/2.0 Receiver Subsystem から出力されるデータは、v\_axi4s\_vid2fifo.v によって、1 クロック遅れて FIFO に書き込まれる。FIFO はある程度のバッファリングが行われるため、一定クロック経過後に empty が立ち下がり、データが読み取れる状態となる。v\_axi4s\_fifo2eth.v によって、empty の立ち下がりの 1 クロック遅れで、Ethernet、IP、UDP ヘッダーの生成を行う。ヘッダーの生成中にも映像データが FIFO にたまり続ける。ヘッダーの生成がおわる 1 クロック前に rd\_en を立ち上げ、FIFO のデータを読み取る。UDP データとして映像データを書き込み、almost\_empty の立ち下がりで rd\_en を立ち下げる。

図 3.12 と図 3.13 は、本実装を稼働させたときの ILA<sup>1</sup> と呼ばれる FPGA の内部信号をモニターするためのツールを使った際に、HDMI の入力と出力を検証した様子である。

図 3.12 は、10 Gigabit Ethernet Subsystem のクロックドメインでのデータの様子である。v\_axi4s\_fifo2eth.v で送信用の FIFO からデータを読み取り、実際にデータを送信し、折り返し受信したデータを v\_axi4s\_eth2fifo.v で受信用の FIFO に書き込む状態である。実際に送受信をしているため、数クロック程度の遅延があることが確認できる。

図 3.13 は、Video Processing Subsystem のクロックドメインでのデータの様子である。前述の通り、almost\_empty の立ち下がりを合図に、パケットを生成してから映像データを書き込むまでに一定のクロックが経過するため、FIFO への書き込みがバッファリングされる。hdmi\_rx\_tvalid が頻繁に立ち上がりと立ち下がりを繰り返しているのに対し、hdmi\_tx\_tvalid はある程度まとまった周期で立ち上がりをしている様子が確認できる。

最後に、表 3.4 に、論理合成後の FPGA のリソース使用状況について示す。

<sup>1</sup>Integrated Logic Analyzer

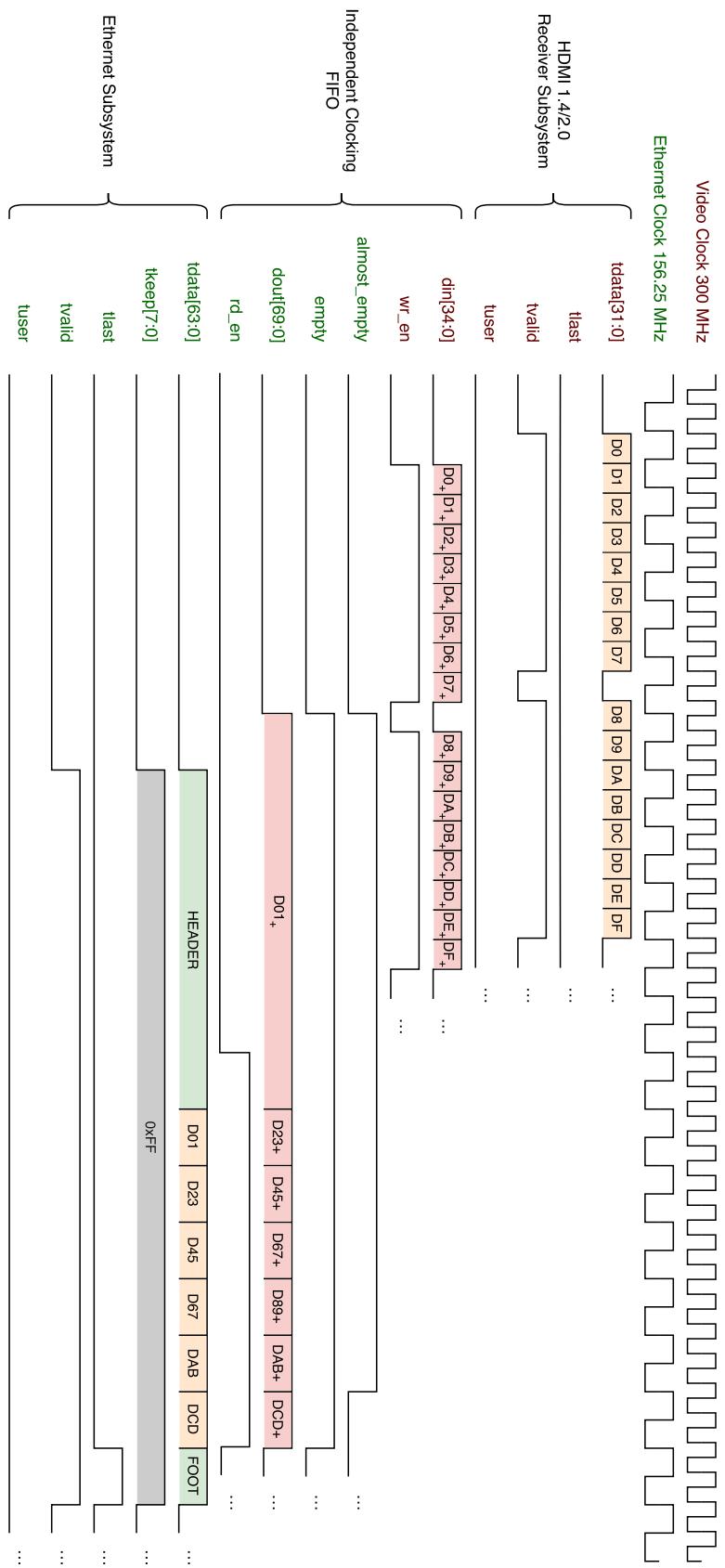


図 3.11: 2つのクロックによるモジュール間でのデータ伝送の波形イメージ

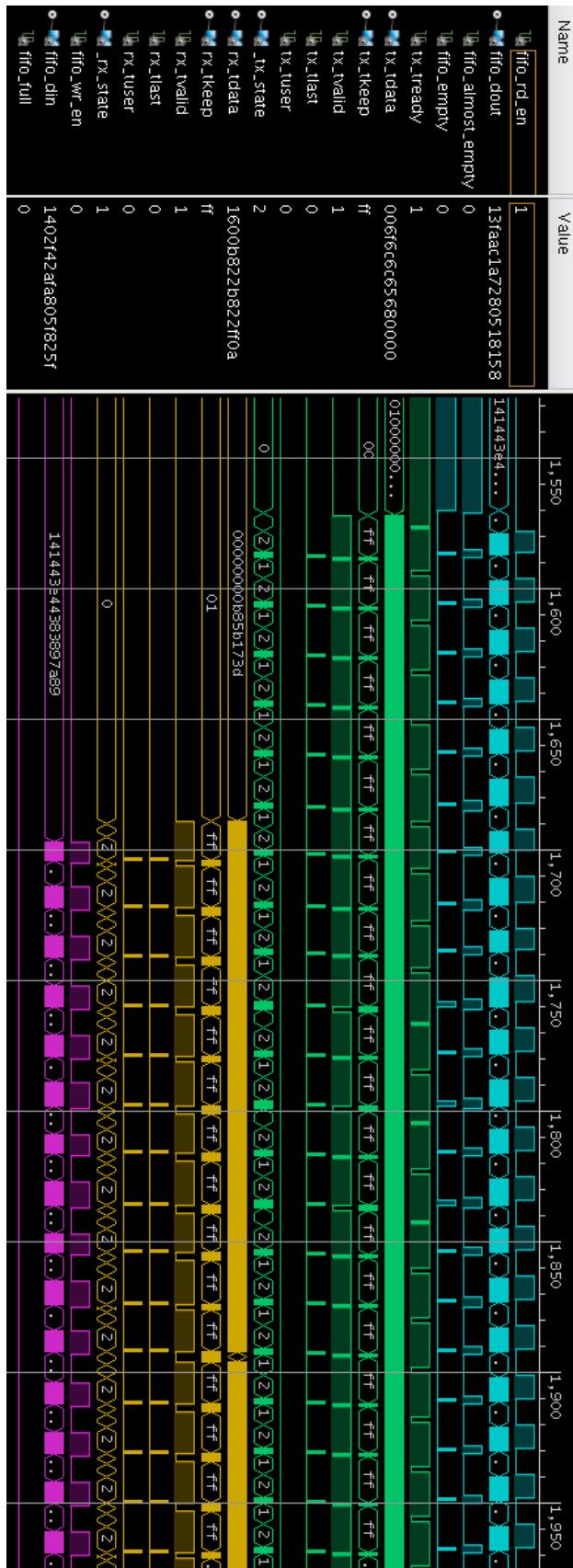


図 3.12: ILA による IP 伝送時の FIFO と Ethernet Sub-system のデータのダンプ

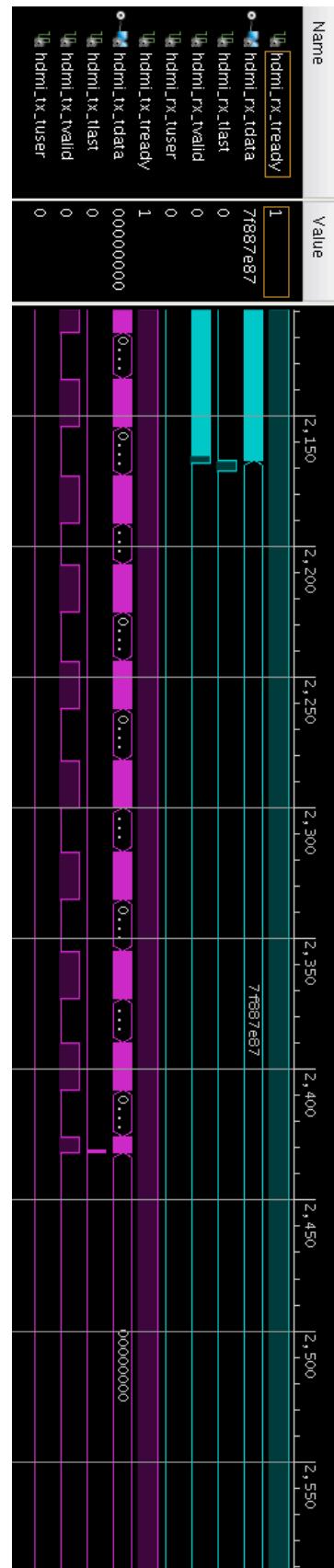


図 3.13: ILA による IP 伝送時の HDMI 入出力のデータのダンプ

表 3.4: 論理合成後のリソース使用状況

リソース	使用	全体	使用率
LUT	48474	203800	23.79%
LUTRAM	4696	64000	7.34%
FF	55768	407600	13.68%
BRAM	310.50	445	69.78%
DSP	23	840	2.74%
IO	40	500	8.00%
GT	4	16	25.00%
BUFG	20	32	62.50%
MMCM	3	10	30.00%
PLL	1	10	10.00%

### 3.3 まとめ

本章では、映像制作現場における IP 伝送装置の要件について評価するため、4K 映像を非圧縮 IP 伝送をするシステムをソフトウェアとハードウェアで設計と実装を行った。ハードウェア実装とソフトウェア実装で互換性のある設計にはなっていないため、相互を組み合わせることはできない。

次章では、これらの実装を実際に評価する。

# 第4章 評価

2章で述べた、映像制作現場におけるIP伝送装置の要件に基づき、3.1章でソフトウェア、ハードウェアそれぞれ実装を行った。本章では、ソフトウェアとハードウェアによる実装が、映像制作現場におけるIP伝送装置の要件を満たしているかを評価する。具体的には、トラフィック、遅延、解像度の対応の3つの項目において、その計測手法と計測結果をまとめ、考察を述べる。

## 4.1 トラフィック

本節では、実装したIP伝送装置が理想通りにパケットの送信を行っているかを確認するために、Linux PCを用いてトラフィックを計測する。

理想的な1秒あたりの映像データのバイト数は、垂直ピクセル $p_w$ 、水平ピクセル $p_h$ 、ピクセルあたりのビット数 $b$ 、フレームレート $f$ を用いて、次のように求められる。

$$p_w \times p_h \times \frac{b}{8} \times f$$

本計測で入力する映像データは、4K 30P のYCbCr 4:2:2 のデータとする。理想的な1秒あたりの映像データのバイト数は、次のように求められる。

$$3840 \times 2160 \times 2 \times 30 = 497,664,000$$

本計測では、計測した1秒あたりの受信バイト数が、理想的な1秒あたりの映像データのバイト数と同値であるかを調べる。

### 4.1.1 計測手法

ソフトウェア実装の場合は受信PC、ハードウェア実装の場合はIP分配折り返しボードに接続されたデバッグ用のPCで、受信バイト数、受信パケット数、破棄パケット数の統計情報を計測した。ネットワークインターフェースの情報は/proc/net/devを監視した。rrdtoolを使い集計し、グラフとして出力した。

#### 4.1.2 計測結果

ソフトウェア実装における伝送中の受信バイト数のグラフを図 4.1 に、受信パケット数と破棄パケット数のグラフを図 4.2 に示す。縦軸は 1 秒あたりのバイト数、またはパケット数を表し、横軸は時間軸を表している。

図 4.1 では、受信バイトの平均は 489MBps となっており、理想的な 1 秒あたりの受信バイト数を満たしていない。この原因としては、送信 PC がデータの送信に追いつかない場合に自動的にフレームをドロップさせる処理によるものだと考えられる。

ハードウェア実装における伝送中の受信バイト数のグラフを図 4.3 に、受信パケット数と破棄パケット数のグラフを図 4.4 に示す。縦軸は 1 秒あたりのバイト数、またはパケット数を表し、横軸は時間軸を表している。

図 4.4 で示した受信パケットのグラフでは、パケットがドロップしている事が確認できる。これは、カーネルのネットワーク処理が、FPGA のハードウェアによる出力の速度に追いつけなかったためだと考えられる。ここでは、グラフから得られる情報を元に、本来 PC が受信したであろうバイト数を計算によって導き出し、評価を行う。

カーネルが処理した 1 パケットあたりの平均バイト数  $B_{avr}$  は、受信したバイト数  $B_{rcv}$ 、受信したパケット数  $P_{rcv}$  を用いて、次のように求められる。

$$B_{avr} = \frac{B_{rcv}}{P_{rcv}}$$

本来 PC が受信したであろうバイト数  $B_{all}$  は、1 パケットあたりの平均バイト数  $B_{avr}$ 、受信したパケット数  $P_{rcv}$ 、破棄したパケット数  $P_{drp}$  を用いて、次のように求められる。

$$B_{all} = (P_{rcv} + P_{drp}) \times B_{avr}$$

パケットには、ヘッダーとフッターが 56bytes 存在しているため、受信バイトのうち有効データ率  $R_{vld}$  は、1 パケットあたりの平均バイト数  $B_{avr}$  を用いて、次のように求められる。

$$R_{vld} = 1 - \frac{56}{B_{avr}}$$

このとき、1 秒あたりの映像データのバイト数  $B_{video}$  は、次のように求められる。

$$B_{video} = R_{vld} \times B_{avr}$$

図 4.3 と図 4.4 から読み取った値から、次のように求められる。

$$B_{avr} = \frac{694,800,066.36}{5,437,845.10} = 127.77$$

$$B_{all} = (5,437,845.10 + 1,494,269.10) \times 127.77 = 885,716,231$$

$$R_{vld} = 1 - \frac{56}{127.77} = 0.5617$$

$$B_{video} = 885,716,231 \times 0.5617 = 497,506,806$$

$B_{video}$  は、理想的な 1 秒あたりの映像データのバイト数とおおよそ一致している。

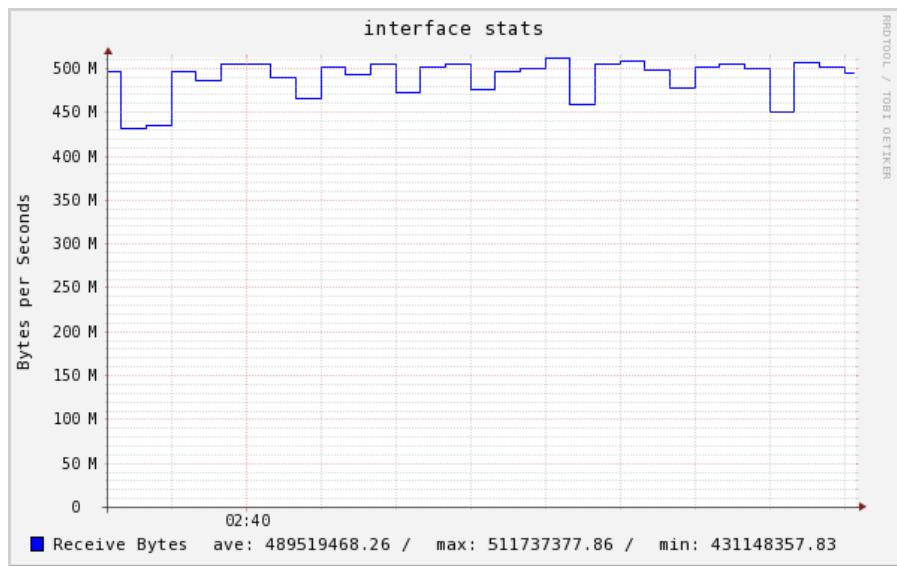


図 4.1: ソフトウェア実装における伝送中の受信バイトのグラフ

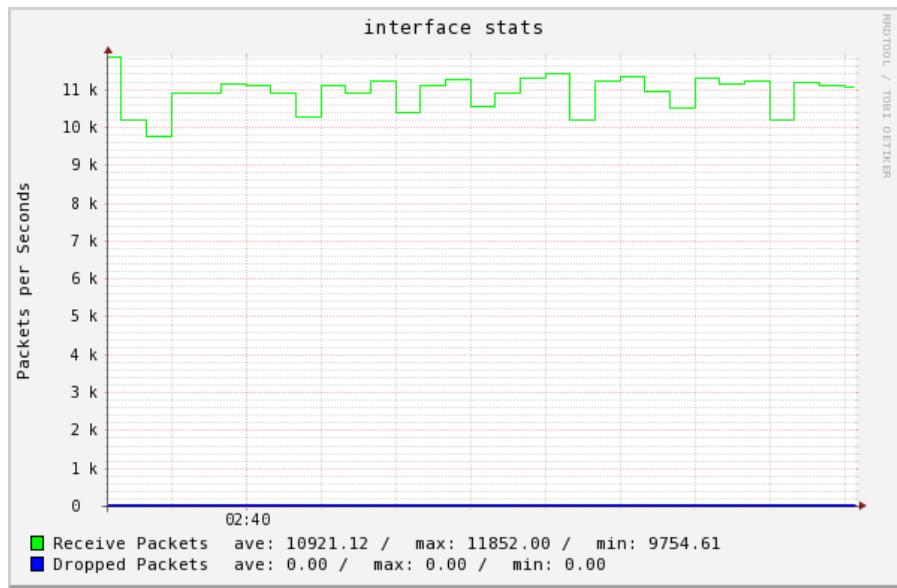


図 4.2: ソフトウェア実装における伝送中の受信パケットのグラフ

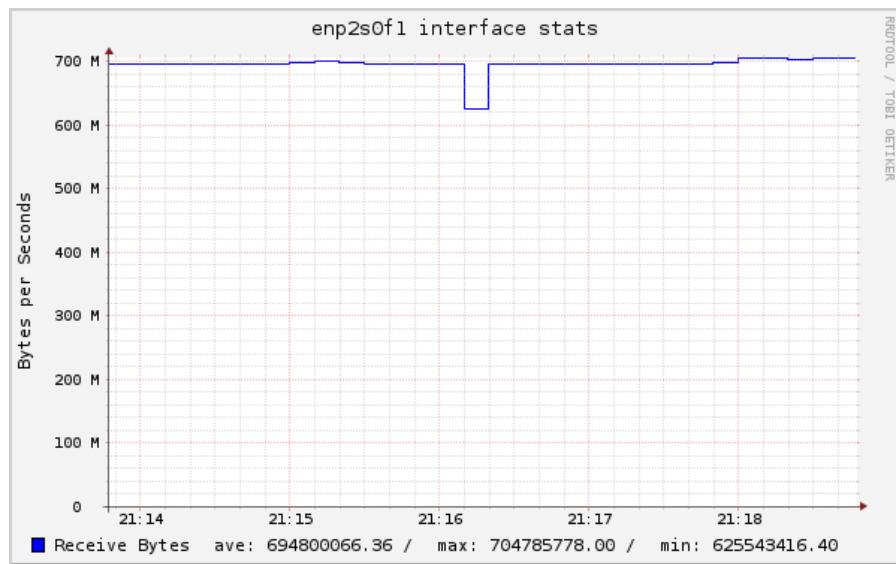


図 4.3: ハードウェア実装における伝送中の受信バイトのグラフ

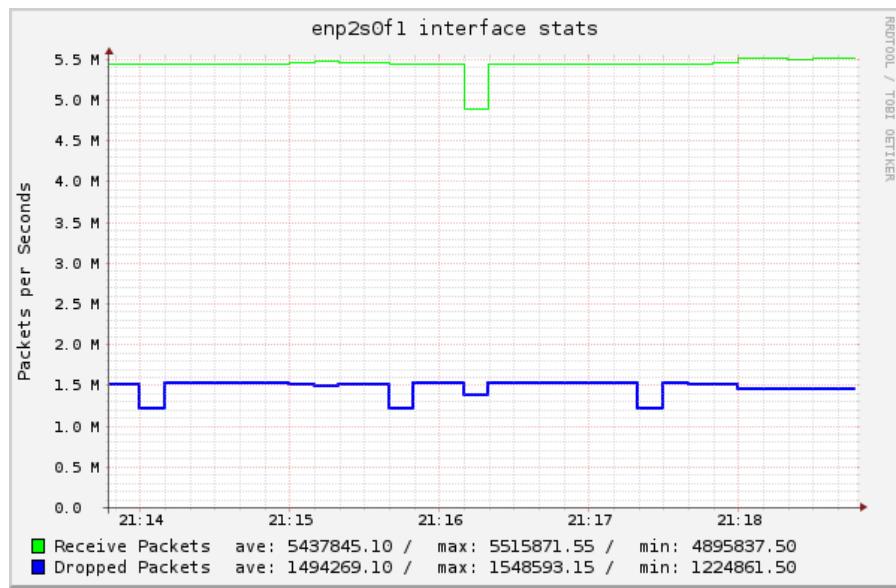


図 4.4: ハードウェア実装における伝送中の受信パケットのグラフ

#### 4.1.3 考察

ソフトウェア実装では、平均 1 万パケットであるのに対し、ハードウェア実装では 700 万パケットもあることが、トラフィックの計測でわかった。これは、3.1 章で述べたハードウェア実装で FIFO のデータが空でなければパケットを生成し、送信するためである。有効データ率も 0.5617 と非常に少なく、非効率である。パケットの生成において、バッファ区間を増やす、一定の映像データが溜まつたらパケットを生成するなど、より効率化しなければならない事がわかった。

## 4.2 遅延

本節では、実装した IP 伝送装置が映像制作現場において許容できる遅延の範囲内かどうかを検証するために、遅延を計測する環境を用意し、発生する遅延を計測した。本計測で入力する映像データは、4K 30P の YCbCr 4:2:2 のデータとする。

#### 4.2.1 計測手法

映像機器の遅延を計測するため、テスト信号生成、マルチビュー生成、画面キャプチャーの機能を有する機器を用意した。遅延を計測した機器の構成を図 4.5 に示す。

テスト信号生成では、フレーム単位のタイムコードが表示された同じソースの映像を 2 つの信号として出力する。一方をスイッチャーへ入力し、もう一方を検査対象となる機器に入力し、その出力をスイッチャーへ入力する。これにより、2 つの信号の遅延は、検査対象となる機器で発生した遅延に抑えることができる。スイッチャーに入力された 2 つの信号はマルチビューとして 1 つの画面に表示され、その画面をキャプチャーすることにより、ある瞬間の 2 つの信号を 1 つの画像で確認することができる。このスイッチャーには、フレーム同期の機能があり、1 フレームより短い期間でバッファリングされるため、計測できる粒度はフレーム単位となる。

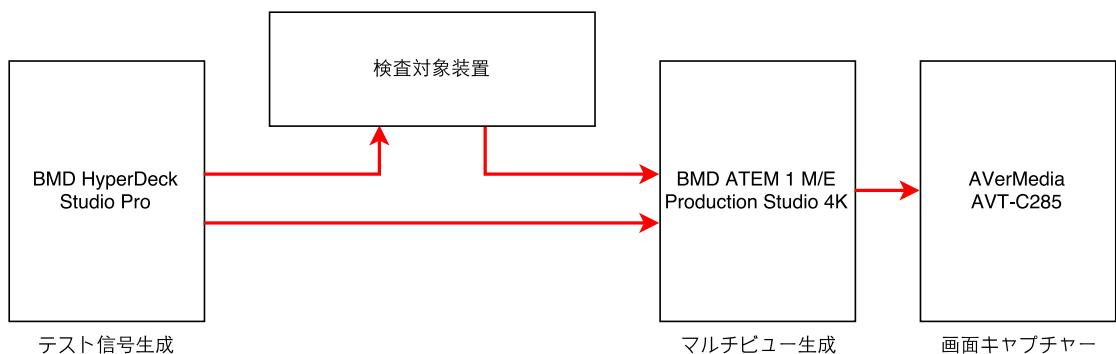


図 4.5: 遅延の計測手法

キャプチャーした瞬間によっては、IP 伝送装置のタイミングにより遅延のばらつきが出る可能性があるため、5 回計測を行う。

#### 4.2.2 計測結果

ソフトウェア実装による遅延時間の計測の様子を図 4.6 と図 4.7 に示す。また、計測結果を表 4.1 に示す。ハードウェアよりも遅延が多く、計測回数によってばらつきがあることが分かる。遅延フレームの平均は 6 フレームとなり、30FPS では 199.99ms となる。

表 4.1: ソフトウェア実装による 30FPS における遅延時間の計測結果

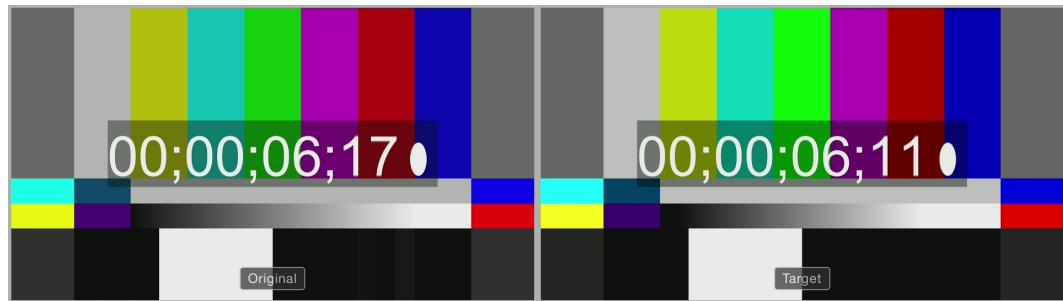
計測回数	遅延フレーム
1 回目	6 フレーム
2 回目	6 フレーム
3 回目	3 フレーム
4 回目	7 フレーム
5 回目	9 フレーム
平均	6 フレーム

ハードウェア実装による遅延時間の計測の様子を図 4.8 に示す。遅延時間の計測は 5 回行ったが、遅延フレームはすべて 0 フレームであった。そのため、計測結果の表については省略する。計測環境の制限から遅延は 1 フレーム以内であるという結果になった。

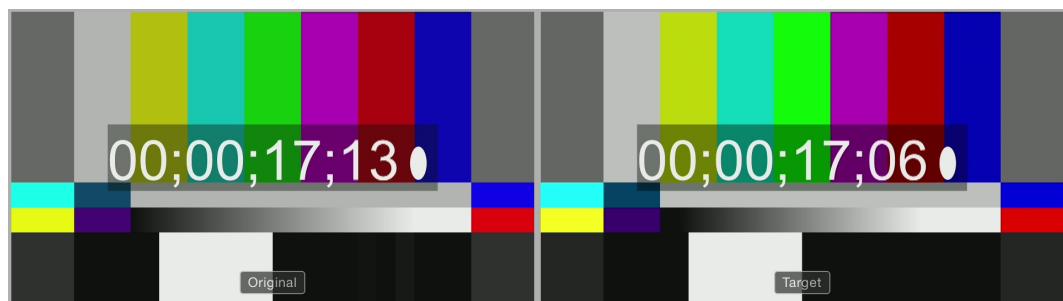
#### 4.2.3 考察

計測の結果、ソフトウェア実装では平均 199.99ms の遅延、ハードウェア実装では 33.33ms 以内の遅延であった。2 章で述べた、映像制作現場における IP 伝送装置の要件では、133.33ms の遅延を許容できるとしている。ソフトウェア実装では要件である 133.33ms の遅延よりも多く、映像制作現場に適さないことが分かる。一方、ハードウェアによる実装では、33.33ms 以内の遅延であり、映像制作現場における IP 伝送装置として優位であることが分かる。

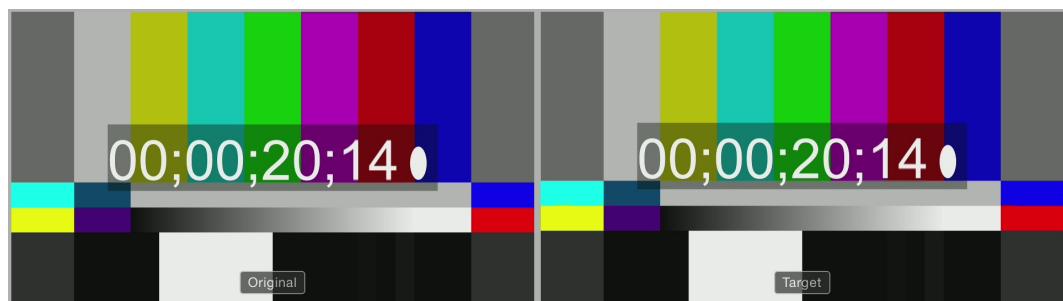
計測から判明したこととして、ソフトウェア実装では計測回数によって遅延フレームにはばらつきが出た。これは、出する映像が常に一定のフレームレートで出力していないということである。一方、ハードウェア実装では 33.33ms 以上の間隔のばらつきは出ていない。この点においても、ハードウェア実装のほうが優れていると言える。



左がオリジナルの信号、右が検査対象の信号  
図 4.6: ソフトウェア実装による 1 回目の遅延計測のキャプチャー画像



左がオリジナルの信号、右が検査対象の信号  
図 4.7: ソフトウェア実装による 4 回目の遅延計測のキャプチャー画像



左がオリジナルの信号、右が検査対象の信号  
図 4.8: ハードウェア実装による遅延計測のキャプチャー画像

## 4.3 解像度

本節では、実装した IP 伝送装置が映像制作現場における IP 伝送装置の要件となる解像度である、4K 30P と 4K 60P の映像伝送に対応しているかどうかを計測した。

### 4.3.1 計測手法

実装した IP 伝送装置に HDMI 2.0 を用いた YCbCr 4:2:0 の出力に対応したカメラを接続し、各解像度で最終的に出力まで確認ができるかを目視で調査した。YCbCr 4:2:0 の出力に対応しているカメラとして SONY FDR-AX1 を使用した。



図 4.9: YCbCr 4:2:0 対応の 4K カメラ FDR-AX1

### 4.3.2 計測結果

ソフトウェア実装では、FDR-AX1 を接続した状態で、4K 30P の出力が確認できたが、4K 60P の出力を確認することができなかった。これは、キャプチャーボードの制約により 4K 30P までの対応であるためである。ハードウェア実装では、4K 30P と 4K 60P ともに正常に出力が確認できた。

その様子を図 4.10 に示す。また、計測結果を表 4.2 に示す。

表 4.2: 実装した伝送装置による YCbCr 4:2:0 の対応状況

	ソフトウェア実装	ハードウェア実装
4K 30P	対応	対応
4K 60P	ハードウェア制約により検証できず	対応



図 4.10: ハードウェア実装による YCbCr 4:2:0 での出力の様子

### 4.3.3 考察

ソフトウェア実装では、4K 30P での伝送は行えたが、キャプチャーボードによる制約で 4K 60P での伝送を行なうことができなかった。ソフトウェア的な制約ではなく、あくまでもハードウェアの環境依存であるため、4K 60P に対応したキャプチャーボードがあれば 4K 60P の伝送が可能であると考えられる。ハードウェア実装では、4K 30P と YCbCr 4:2:0 による 4K 60P の伝送を行なうことができることが実際に確認できた。

前述の通りソフトウェア実装では、ソフトウェア要因による制約は確認できなかつたため、ソフトウェア実装の YCbCr 4:2:0 による伝送についてハードウェアの環境を再考する必要がある。

## 4.4 まとめ

ハードウェアによる実装では、YCbCr 4:2:0 による 4K 60P の伝送が可能であることに加え、遅延も 33.33ms 以内であり、映像制作現場において優位に扱えることがわかった。一方、ソフトウェアによる実装では、遅延が平均で 6 フレームとなり、映像制作現場における要件である 133.33ms 以内に抑えることができなかつた。

# 第5章 結論

## 5.1 本研究のまとめ

本研究では、映像制作現場におけるIP伝送の優位性を示すために、現状の映像伝送における必要条件と課題点を洗い出し、IP伝送でその必要条件を満たすことができ、さらに課題点をクリアすることができるかを証明した。

まず、映像伝送における重要な要件の1つである遅延の許容範囲を満たすために、映像制作現場においてどれほどの遅延が許容できるのかを調査した。平均166.66msの遅延があると、許容できないという被験者が2割弱となった。この調査から、IP伝送では166.66ms以内の遅延に抑えるべきと結論づけた。

次に、4K映像をIP伝送するシステムを、ソフトウェアとハードウェアで実装し、それについて要件が満たせるかについて検証と評価を行った。

ソフトウェアによる実装では、汎用的なPCでもキャプチャーボードと10GbpsのNICがあれば4K映像のIP伝送が行えることがわかった。評価では、遅延が平均で6フレーム、すなわち199.99ms発生し、映像制作現場における遅延の要件を満たすことができなかった。

ハードウェアによる実装では、XilinxのKC705とTEDのHDMIインターフェースカードを使い、EthernetとHDMIのデータの変換を行う回路を設計した。評価では、遅延は1フレーム以内であり、映像制作現場における遅延の要件を満たすことができた。

## 5.2 本研究の結論

ハードウェアの実装では、最初に示した映像伝送における重要なポイントうち、遅延を一定以下にし稼働することが可能であった。また、YC b Cr 4:2:0方式により60Pでの伝送も可能であった。

これらの点をまとめると、映像伝送における必要な要件が満たせ、映像制作現場におけるIP伝送の優位性が立証できた。

## 5.3 今後の課題と展望

本研究では、IP伝送による映像配信システムの設計と実装を行った。

ソフトウェアによる実装では、キャプチャーボードを使用したため、キャプチャーと描画に最低でも 2 フレームの遅延が生じてしまう。また、ソフトウェア制御であるため動作の安定も保証できない。ソフトウェアでの传送は、TCP レイヤー上で传送した。本来、映像传送あれば、UDP レイヤー上で実装するべきであるが、順序制御、再送制御などの実装を省くためであった。

ハードウェアによる実装では、FIFO ベースでのパケットを生成するため、UDP レイヤー上の独自のプロトコルで 1 秒あたり 700 万パケットを生成していた。これが効率の良い IP 传送方式ではないため、既存の IP 传送規格に合わせ実装したい。

また、ハードウェアによる実装では、クロックを同じハードウェアで共有しているため、クロックについての考慮をしていない。そのため、他のハードウェアでもクロック情報を共有する必要がある。

## 謝辞

本研究を進めるにあたり、ご指導いただきました慶應義塾大学 環境情報学部教授 村井純博士、同学部教授 中村修博士、同学部准教授 Rodney D. Van Meter III 博士、同学部准教授 植原啓介博士、同学部准教授 中澤仁博士、SFC 研究所 上席所員（訪問）斎藤賢爾博士に感謝致します。

研究について日頃からご指導頂きました松谷健史博士、空閑洋平博士、理工学研究科開放環境科学専攻 後期博士課程 德差雄太氏に感謝致します。研究室に所属したばかりの頃から本研究に至るまで、特定の分野にこだわらない広い視点で何年生の時であっても妥協のない姿勢で向かい合い、絶えず多くのご指導をいただきました。本研究を卒業論文としてまとめることができたのも両氏のおかげです。重ねて感謝申し上げます。

本研究の評価に必要な伝送装置の助言、機材を運搬していただいた一般社団法人 Mozilla Japan 工藤紀篤博士に感謝いたします。評価に必要な伝送装置を借用させていただいた慶應義塾大学デジタルメディア・コンテンツ統合研究センターの皆様に感謝いたします。長期の間、開発、実験用に 4K カメラなどの機器を借用させていただいた慶應義塾大学湘南藤沢メディアセンターマルチメディアサービスの皆様に感謝いたします。実証実験を行った ORF2015 では、実行委員会の皆様、ネットワーク環境を整備していただいた ITC の皆様、ORF NOC の皆様、映像制作をしていただいた音像工房の皆様に感謝いたします。

研究室を通じた生活の中で多くの示唆を与えてくれた阿部涼介氏、沖幸太朗氏、木下舜氏、黒米祐馬氏、鈴木恒平氏、高橋佑允氏、原雅彦氏、細田航星氏、および Arch 研究グループの皆様に感謝します。また、徳田・村井・楠本・中村・高汐・バンミーター・植原・三次・中澤・武田 合同研究プロジェクトの皆様に感謝致します。

最後に、私の研究を支えてくれた両親をはじめとする親族、多くの友人・知人に感謝し、謝辞と致します。



# 参考文献

- [1] 10 Gigabit Ethernet Subsystem v3.0 (PG157). [https://www.xilinx.com/support/documentation/ip\\_documentation/axi\\_10g\\_ethernet/v3\\_0/pg157-axi-10g-ethernet.pdf](https://www.xilinx.com/support/documentation/ip_documentation/axi_10g_ethernet/v3_0/pg157-axi-10g-ethernet.pdf).
- [2] Big Buck Bunny. <http://www.bigbuckbunny.org/>. Copyright (C) 2008 Blender Foundation — peach.blender.org Some Rights Reserved. Creative Commons Attribution 3.0 license.
- [3] FIFO Generator v12.0 (PG057). [https://www.xilinx.com/support/documentation/ip\\_documentation/fifo\\_generator/v12\\_0/pg057-fifo-generator.pdf](https://www.xilinx.com/support/documentation/ip_documentation/fifo_generator/v12_0/pg057-fifo-generator.pdf).
- [4] HDMI 1.4/2.0 Receiver Subsystem v2.0 (PG236). [https://www.xilinx.com/support/documentation/ip\\_documentation/v\\_hdmi\\_rx\\_ss/v2\\_0/pg236-v-hdmi-rx-ss.pdf](https://www.xilinx.com/support/documentation/ip_documentation/v_hdmi_rx_ss/v2_0/pg236-v-hdmi-rx-ss.pdf).
- [5] HDMI 1.4/2.0 Transmitter Subsystem v2.0 (PG235). [https://www.xilinx.com/support/documentation/ip\\_documentation/v\\_hdmi\\_tx\\_ss/v2\\_0/pg235-v-hdmi-tx-ss.pdf](https://www.xilinx.com/support/documentation/ip_documentation/v_hdmi_tx_ss/v2_0/pg235-v-hdmi-tx-ss.pdf).
- [6] HDMI 2.0 Implementation on Kintex-7 FPGA GTX Transceivers. [https://www.xilinx.com/support/documentation/application\\_notes/xapp1287-hdmi-on-fpga-gtx-transceivers.pdf](https://www.xilinx.com/support/documentation/application_notes/xapp1287-hdmi-on-fpga-gtx-transceivers.pdf).
- [7] HDMI 2.0 カード TB-FMCH-HDMI4K. <http://www.inrevium.com/product/video/index.html>.
- [8] KC705 Evaluation Board for the Kintex-7 FPGA. [https://www.xilinx.com/support/documentation/boards\\_and\\_kits/kc705/ug810\\_KC705\\_Eval\\_Bd.pdf](https://www.xilinx.com/support/documentation/boards_and_kits/kc705/ug810_KC705_Eval_Bd.pdf).
- [9] miyukki/mv-delay-measurement. <https://github.com/miyukki/mv-delay-measurement>.
- [10] sfc-arch/bmd-4k-streaming. <https://github.com/sfc-arch/bmd-4k-streaming>.
- [11] Video PHY Controller v2.0 (PG230). [https://www.xilinx.com/support/documentation/ip\\_documentation/vid\\_phy\\_controller/v2\\_0/pg230-vid-phy-controller.pdf](https://www.xilinx.com/support/documentation/ip_documentation/vid_phy_controller/v2_0/pg230-vid-phy-controller.pdf).

- [12] Sony Business Solutions Corporation. ネットワーク・メディア・インターフェース.  
[https://www.sony.jp/products/Professional/c\\_c/nmi/](https://www.sony.jp/products/Professional/c_c/nmi/).
- [13] L. Gharai and C. Perkins. RTP Payload Format for Uncompressed Video. RFC 4175, Internet Engineering Task Force, September 2005.
- [14] HDMI Licensing, LLC. *High-Definition Multimedia Interface Specification*. Version 1.4.
- [15] HDMI Licensing, LLC. *High-Definition Multimedia Interface Specification*. Version 2.0.
- [16] Blackmagic Design Pty. Ltd. Blackmagic Desktop Video SDK. <https://www.blackmagicdesign.com/jp/support/family/capture-and-playback>.
- [17] Inc. NewTek. NDI. <http://www.newtek.com/ndi.html>.
- [18] Society of Motion Picture and Television Engineers. *SMPTE 2022*.
- [19] Society of Motion Picture and Television Engineers. *SMPTE 2110*.
- [20] Society of Motion Picture and Television Engineers. *SMPTE RDD 34*.
- [21] Society of Motion Picture and Television Engineers. *SMPTE ST 2081-1*.
- [22] Society of Motion Picture and Television Engineers. *SMPTE ST 2082-1*.
- [23] 小寺信良. 【小寺信良の週刊 Electric Zooma!】コンテンツのHDR化、IP伝送による作り手側の混乱。InterBEEで見た理想と現実- AV Watch. <http://av.watch.impress.co.jp/docs/series/zooma/1033618.html>.
- [24] 小寺信良. 【小寺信良の週刊 Electric Zooma!】第733回：裏方の大革命、4K放送に向け、“IP伝送”的道筋が見えてきた「InterBEE 2015」- AV Watch. <http://av.watch.impress.co.jp/docs/series/zooma/732055.html>.