

フロッピーディスクを利用したDV送受信システムの構築

千代 佑[†] 小川 晃通[‡] 杉浦 一徳^{*} 中村 修[†] 村井 純[†]

慶應義塾大学環境情報学部[†] 慶應義塾大学大学院政策・メディア研究科[‡] 独立行政法人通信総合研究所^{*}

本研究では、DV 転送システムである Digital Video Transport System (以下、DVTS) を単一フロッピーディスクから起動し、DV データの送信 (dvsend) 及び DV データの受信 (dvrecv) を可能とする機構を構築した。本機構では、DVTS 起動において必要最小限の機能を組み込んだ Linux Kernel, 及び圧縮したファイルシステムを使用し、これらを 1 枚のフロッピーディスク上で動作するシステムとして実装した。ルートファイルシステムは起動時に RAMDISK に展開した。また、ファイルシステム内には DVTS (dvsend, dvrecv) の他に IP アドレスを設定するツールやネットワークの状態を確認するツール等を格納した。本機構により、既存のハードディスクによる環境構築を行う必要がなくなり DV 送信・受信の実現が可能となった。また、RAMDISK 上にファイルシステムを展開することにより、従来の UNIX 環境と同等の操作を行う必要なく電源を切ることが可能となった。

Construction of DV Transceiver System using the Floppy Disk

Tasuku Chiyo[†] Akimichi Ogawa[‡] Kazunori Sugiura^{*}

Osamu Nakamura[†] Jun Murai[†]

[†] Faculty of Environmental Information, Keio University

[‡] Graduate School of Media and Governance, Keio University

^{*} Communications Research Laboratory

The purpose of this research is to develop Digital Video Transport System (following, DVTS) based on bootable floppy media. In this research, a single floppy enables DV transmission (dvsend) and DV receive (dvrecv). With this mechanism, Linux Kernel, in which the minimum of DVTS starting function is included, and compressed file system are used, and they are mounted as a system and operated on one floppy disk. The root file system is uncompressed to RAMDISK at the time of booting. Besides DVTS (dvsend and dvrecv), this system stores tools which can set up the IP address and also check the state of a network. With the use of this mechanism, the existing hard disk environment is no longer required in order to realize DV transmission and reception. Thus, by uncompressing this system on RAMDISK, the power supply is safely shut off without operation equivalent to the conventional UNIX environment.

1 はじめに

1.1 背景

Digital Video Transport System[3](以下、DVTS)はDV[1]データを転送するシステムとして有用性が高い。しかし、現状ではハードウェアがDVTS動作

のための必要条件を満たしている場合でも、Linux、Windows等なんらかの既存の環境上にDVTSをインストールしシステムを構築する必要がある。このため、DVTSを利用する必要性が生じた場合、既存の環境に変更を加える必要がある。

しかし、実際は利用者の都合により既存の環境に変更を加える事が容易ではなかったり、不可能な場合があったりする。このような場合、新規にPCを用意する等してDVTSを利用するための環境を構築する必要がある。

^{††}Keio University Shonan Fujisawa Campus
5322, Endo, Fujisawa, Kanagawa 252, Japan
E-Mail: {tsk, akimichi, osamu, jun}@sfc.wide.ad.jp
^{*}4-2-1 Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan
E-Mail: uhyo@sfc.wide.ad.jp
Members of Murai LAB, those in especially STREAM

また、DVTSを動作させるために構築した環境を不意の電源断やハードディスクドライブ(以下、HDD)の故障等のトラブルによって失う事も考えられる。これらは利用者にとって新たな労力や費用を要する作業となる。

1.2 DVTSとは

Digital Video Transport System(略称、DVTS)とはインターネットを利用して、DVデータを送受信するアプリケーションである。約30Mbpsの帯域を利用し、TVと同等の品質が実現されている。

これまでFreeBSDへの実装が行われていたが、現状ではFreeBSDの他にLinux、MacOS X、NetBSD、Windows2000、WindowsXPで動作する。

2 研究概要

本研究の目的はDVTSの可搬性を追求し、利用者が既存の環境に変更を加えることなく容易にDVTSを利用できる環境の実現である。ユーザー環境に広く普及しているフロッピーディスク(以下、FD)を利用し、DVTS専用のFDブートシステムを構築する事を提案する。

DVTS専用のシステム構築のための基盤となるオペレーティングシステム(以下、OS)としてLinuxを用いた。フロッピーによって起動するLinux環境にて動作するDVTSを組み込んだ。本実装により、ハードウェアがDVTS動作のための必要条件を満たしていれば、FD1枚でDV送受信を行うことが可能となる。

前述のシステムを用いれば、PCのハードディスクにある既存のユーザー環境に変更を加えることなく容易にDV送受信を行うことが可能となる。そこで、1枚のFDにDVTS専用のLinux環境を構築する事を本研究の目的達成のための手法の目標とした。

3 設計

本システムの設計目標として以下の3点をあげる。

- 本機構利用者のDVTS動作のためのOS環境依存性の解消
- FD1枚で動作可能な可搬性
- 利用者にとって容易なDVTS動作環境の構築

3.1 設計概要

本機構は、DV Sender、DV Receiver 双方として利用可能である。本機構利用者はFDからのカーネルブート後、そのPCをDV Sender、DV Receiver どちらとして利用するかを選択する。本設計では、1対1

でのDVデータ通信の他、IPマルチキャストを利用した1対nのDVデータ通信を行う事も可能である。

図1に本システムの設計概要図を示す。

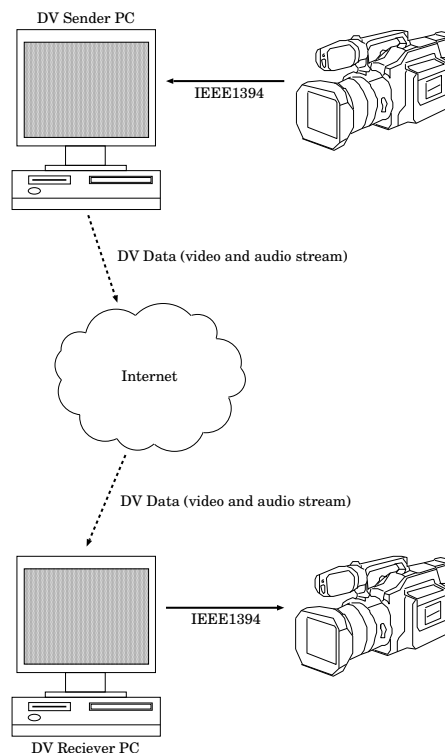


図1: システム設計概念図

動作手順は以下の通りになる

1. DVカメラ等のIEEE1394[2]デバイスを接続したDV SenderとなるPCを、本システムを利用して起動、設定
 2. DV Sender PCからDVデータの送信
 3. DV Sender PCからインターネットを介し、DV Receiver PCにDVデータの転送
 4. DV Receiver PCでDVデータを受信
 5. 受信したDVデータをDV Receiver PCに接続したIEEE1394デバイス(DVカメラ等)に出力
- 設計の要件として以下の項目が挙げられる
- オペレーティングシステム
 - カーネル
 - ファイルシステム
 - インターフェース
 - Network Interface Cardのサポート
 - IP Address, Routingの設定
 - IEEE1394 Cardのサポート
 - DVデータ送受信

- IEEE1394 デバイスから入力した DV データの送信 (Sender)
- DV データの受信, 及び IEEE1394 デバイスへの出力 (Receiver)

3.2 オペレーティングシステム

カーネルの改変

本研究の設計目標である, DVTS 動作のための OS 環境依存性の解消と, 可搬性の確保には以下の理由から Linux Kernel を用いる.

Linux Kernel は, サポートするデバイス・機能を選択することが可能である. その構造上, システムにおけるカーネルサイズ (容量) を小さく抑えることが可能である. これは FD という限られた容量のメディア上に本システムを実装する上で有利な条件となる.

ファイルシステム

本システムは, 圧縮したファイルシステムをカーネル起動後, メインメモリ上に RAM ディスクを作成し展開する手法を採用する. これは, 限られたディスク容量を有効にファイルシステムの領域として利用する事が可能だからである.

また, 圧縮ファイルシステムを RAM ディスクに展開して利用する事により, 従来の UNIX 環境と同等の操作を行う必要なく電源を切ることが可能になるという副次的な利点を得ることができる.

インターフェース

本システムは, DVTS(dvsend, dvrecv) 動作のために Network Interface Card, 及び IEEE1394 Card 各インターフェースのサポートを必要とする. これらのデバイスは多種多様なものが利用されているが, 本システムでは利用者の計算機環境において普及率が高いと思われるデバイスをサポートする.

DV データ送受信

DV データ送信プログラム及び受信プログラムは既に Linux で動作が確認されているプログラム (dvsend, dvrecv) を用いる. また, これらを実行する際に必要な共有ライブラリとともに FD に格納する.

4 実装

ベースとなるオペレーティングシステムには, Debian GNU/Linux を利用し, これに UNIX 版 DVTS を組み込み, これらを 1 枚の FD に格納する.

本実装は DVTS の動作条件を満たしている PC を使用した. 図 2 に本機構の実装環境を示す.

PC	PLAT'HOME COMPACT STATION (PentiumIII:450MHz)
ネットワーク	Intel 82558 (Onboard)
IEEE1394	Radius Firewire PCI 2330
OS	Debian GNU/Linux 3.0 Linux Kernel 2.4.18
組み込み ソフトウェア	DVTS 0.9a22 Glibc 2.2.5

図 2: 実装環境

また, 本実装の動作実験環境として以下の 2 台の DV カメラと 2 台の PC を使用した. 図 3 に本機構の DV データ送信実験環境, 図 4 に DV データ受信実験環境を示す.

PC	IBM ThinkPad X22 (PentiumIII:800MHz)
ネットワーク	Intel 82559 (Onboard)
IEEE1394	Ultrabase X2 (Onboard)
DV カメラ	Sony DCR-PC10

図 3: DV データ送信実験環境

PC	EPSON DIRECT Endeavor AT-680C (PentiumIII:550MHz)
ネットワーク	Intel 82559 (Onboard)
IEEE1394	Planex DVX-400P (PCI)
DV カメラ	Panasonic NV-EX21

図 4: DV データ受信実験環境

図 5 に本機構の想定利用者計算機環境を示す (これは DVTS 動作のために必要な計算機環境に準拠している).

カーネル

本機構では, Linux Kernel 2.4.18 を用いて本システムに必要な最小限の機能を組み込んだカーネルを構築した. カーネルに組み込んだ機能の一覧を以下に示す.

- (CONFIG_M386)
 - 全ての i386 プロセッサファミリで動作するカーネルのサポート

PC	IBM PC/AT 互換機 (Intel PentiumIII 以上, 又は その互換 CPU. Transmeta Crusoe TM5x00 動作可)
ネットワーク	Intel 8255x シリーズ 3Com 3c590/900 シリーズ SiS 600/900 シリーズ (Onboard, PCI ボード共に 動作可)
IEEE1394	IEEE1394 PCI ボード (TI 製コントローラ互換)
DV カメラ	(機種依存無)

図 5: 想定利用者計算機環境

- (CONFIG_BINFORMAT_ELF)
 - ELF(Executable and Linkable Format) バイナリのカーネルサポート
- (CONFIG_BLK_DEV_FD, CONFIG_BLK_DEV_RAM)
 - PC フロッピーディスクのサポート
 - RAM ディスクのサポート
- (CONFIG_VT, CONFIG_VT_CONSOLE)
 - 仮想端末デバイスのサポート
 - 仮想端末上でのコンソールのサポート
- (CONFIG_PROC_FS)
 - /proc ファイルシステムのサポート
- (CONFIG_EXT2_FS)
 - 第二拡張 ext2 ファイルシステムのサポート
- (CONFIG_VGA_CONSOLE)
 - VGA テキストコンソールのサポート
- (CONFIG_NET)
 - ネットワークのサポート
- (CONFIG_PCI)
 - PCI bus のハードウェアサポート
- (CONFIG_SYSVIPC)
 - System V IPC のサポート
- (CONFIG_INET, CONFIG_IP_MULTICAST)
 - TCP/IP ネットワーキングのサポート
 - IP マルチキャストのサポート
- (CONFIG_IEEE1394, CONFIG_IEEE1394_OHCI1394)
 - IEEE1394(別名 FireWire) のサポート
 - OHCI(Open Host Controller Interface) のサポート
- (CONFIG_IEEE1394_VIDEO1394, CONFIG_IEEE1394_RAWIO)
 - IEEE1394 Video デバイスのサポート
 - IEEE1394 Raw デバイスのサポート
- (CONFIG_NETDEVICES, CONFIG_NET_ETHERNET, CONFIG_NET_PCI)

- ネットワークデバイスのサポート
- イーサネット (10Mbit または 100Mbit) のサポート
- EISA, VLB, PCI バスおよびオンボードタイプのコントローラサポート
- (CONFIG_MSDOS_PARTITION)
 - MSDOS パーティションのサポート

ルートファイルシステム

本機構のルートファイルシステムは, Linux で最も一般的に利用されている ext2 ファイルシステム¹上に作成した. 実装環境において mke2fs コマンドを用い, FD を ext2 ファイルシステムとしてマウントし, これをベースに以下の作業を行いルートファイルシステムを構築した.

デバイスノード

ルートファイルシステムの構築において必要なデバイスノード (ブロックデバイス, キャラクタデバイス) の実装を mknod コマンドを用いて/dev に行った. 前述の ext2 ファイルシステム作成作業において, デフォルトで 184 個の inode²を利用可能に設定している.

FD のような限定された環境では, inode が枯渇してしまうため, 必要最小限のデバイスノードを作成した. 図 6 に, 実装したデバイスノードの一覧を示す.

console	コンソール端末デバイス
fd[0-1]	フロッピーディスクデバイス
kmem	仮想メモリ空間アクセス用デバイス
mem	(同)
null	標準出力を無効化するヌルデバイス
ram0	圧縮 RAM ディスクマウント用デバイス
raw1394	IEEE1394RAW データ入出力デバイス
tty[0-4]	仮想コンソール用デバイス
random	乱数生成用デバイス
urandom	(同)
video1394	IEEE1394 ビデオ入出力デバイス
zero	ゼロデータ提供デバイス

図 6: 実装デバイスノード一覧

¹second extended filesystem

²UNIX 系 OS におけるファイルシステム管理用構造体, 1file=1inode

RAM ディスク

本機構ではルートファイルシステムを圧縮状態で格納し、起動時に RAM ディスクに展開する手法で FD の容量制限 (1440KB) を超えたサイズのカーネル及びルートファイルシステムの格納を可能にした。

起動時にメインメモリ上に 4096KB の RAM ディスクを作成し、ブロックデバイスとして認識させた。また、ルートファイルシステムの圧縮は、gzip を用いた。

プログラム

本実装では、DVTS の設定と動作のために必要な最小限のバイナリプログラムを格納した。

- DVTS
 - dvsend, dvrecv
- ファイルシステム, シェル, プロセス管理関連
 - mount, ash, ls, ps, stty
- ネットワークの設定及び状態確認
 - ifconfig, route, netstat, ping

共有ライブラリ

本機構では、静的リンクでビルドしたバイナリプログラムを格納せず、共有ライブラリを導入した。これはプログラムが依存するライブラリを共有することで必要ディスク容量の減少、ファイル依存関係の明確化、バイナリプログラム入れ換えの容易等が期待できるからである。

図 7 に本機構における共有ライブラリの依存関係を示す。

ブート・ルートディスク

本機構では作成したカーネル及びルートファイルシステムの圧縮イメージを 1 枚の FD に格納可能であるよう作成した。

図 8 に本機構におけるディスクの構造 (ブートプロセス) を示す。

5 評価

評価は設計目標を元に以下の項目に関して行う。

- 本機構利用者の DVTS 動作のための OS 環境依存性の解消
- FD1 枚で動作可能な可搬性
- 利用者にとって容易な DVTS 動作環境の構築
- 起動時間の比較

```
ld-2.2.5.so (ELF Dynamic Linker, Loader)
  ash, dvsend, dvrecv, ifconfig, ls,
  mount, netstat, ping, ps, route, stty

libc-2.2.5.so (GNU C Library)
  ash, dvsend, dvrecv, ifconfig, ls,
  mount, netstat, ping, ps, route, stty

libnsl-2.2.5.so (Name Services Library)
  dvsend, dvrecv

libproc.so.2.0.7 (System and Process
  ps                Monitoring Utilities)

libpthread-0.9.so (POSIX thread Library)
  ls

libraw1394.so.5.0.0 (Direct access to
  dvsend, dvrecv    IEEE1394 bus)

librt-2.2.5.so (Realtime Ext Library)
  ls
```

図 7: 共有ライブラリ依存関係

5.1 本機構利用者の DVTS 動作のための OS 環境依存性の解消

動作実験に用いた 2 台の PC を、本機構を用いることによりインストールされている既存の OS と全く独立したシステム上で DVTS を動作させる事ができた。

5.2 FD1 枚で動作可能な可搬性

実装した圧縮イメージを用い、FD1 枚で起動し、DVTS 動作に必要な一連の作業を行うことができた。

5.3 利用者にとって容易な DVTS 動作環境の構築

利用者は本研究で実装したディスクイメージを、UNIX 環境の dd 又は Windows 環境の rawrite 等のプログラムを用いて FD に書き込む作業のみで、PC の OS 環境に依存しない DVTS 動作環境を得ることができる。

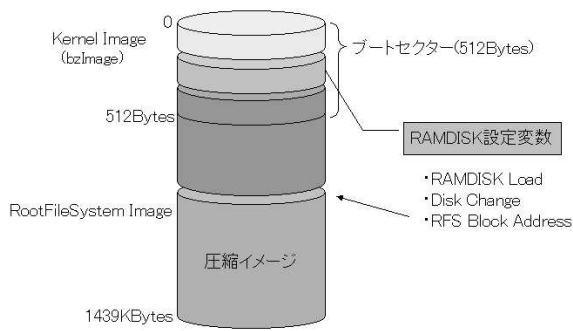


図 8: ディスクの構造

5.4 起動時間の比較

同一の PC を用いて、既存の OS 環境と本機構それぞれ、PC の電源投入から BIOS³起動を経てシステムブートに要する時間を比較した、図 9 に比較した OS 環境とそれぞれのブート時間を示す。

	電源 ON BIOS	BIOS ブート完了
本機構	41.9 秒	45.9 秒
Linux(Debian)	42.1 秒	18.4 秒
Windows2000	42.0 秒	62.2 秒

図 9: システムブートに要する時間

HDD にインストールされた OS において、DVTS 利用のための必要最小限の環境を構築した。同条件の場合、FD は HDD と比較してメディアのリード・ライト速度が遅いため、ブート時間が長かった。しかし、実際の利用者環境においては、多数のデバイスドライバやスタートアップ時に動作させるプログラムが存在する。さらに、通常の Linux においてはシステムブート後、login 認証が存在する。これらを踏まえ、利用者的立場から考えた場合 FD にも利点があると言える。

6 おわりに

6.1 まとめ

本研究では DVTS 利用者にとって可搬性が高く、既存の OS 環境依存を解消した機構を実現した。また、RAM ディスクを用いた設計と実装において、利

用者が不意の電源断や HDD トラブルによって構築済の環境を失わない機構も実現した。利用者は本機構を利用することにより FD1 枚で DVTS を動作させることが可能である。

6.2 今後の展望

本機構は、Linux を用い普及率が高いハードウェアをサポートしたが、より多くの Network Interface Card や PC Card デバイス、USB FD ドライブに対応するために更なるカーネルの改変とドライバ類のモジュール化を行う事が挙げられる。

また、ネットワーク関連では今回、ifconfig, route による手動 IP アドレス設定のみ、また IPv4 のみで実装したが、これらに加えて、DHCP クライアント機能や IPv6 を追加サポートすることで、本機構がより様々な利用環境を想定することが出来る。

参考文献

- [1] *Specifications of Consumer-Use Digital VCRs using 6.3mm magnetic tape*, 1994. HD Digital VCR Conference.
- [2] *IEEE Standard for a High Performance Serial Bus*, 1995. IEEE computer society.
- [3] A.Ogawa. *DVTS (Digital Video Transport System) WWW page*, November 2001. URL:<http://www.sfc.wide.ad.jp/DVTS/>.

³Basic Input/Output System