Video4Linux

Dr. Ralph J.K. Metzler

<rjkm@thp.uni-koeln.de>

- Video4Linux
  - History
  - V4L API and videodev device
  - I2C subsystems
- New Developments
  - Video4Linux2
  - New features and applications
  - MJPEG, MPEG 1/2, DVB, DVD

Overview

- Drivers in early 1997:

  Pro MovieStudio card, FAST Screen Machine II, Cortex I, CX100,

  WinVision B&W, Matrox Meteor, Mutech MV1000 PCI,

  Video Blaster, VT1500 TV tuner, Bt848 (bttv)

  Many drivers for video decoder cards but **no** standard interface.

  All drivers came with their own incompatible application.

Overview

In late 1997 Alan Cox introduced the Video4Linux standard to the 2.1.x kernels.

- Video4Linux drivers now available (in the kernel alone):
  - Mediavision Pro Movie Studio
  - Connectix QuickCam (b/w and color version)
  - ZR36057/36060 (Buz, LML33)
  - ZR36120
  - Bt848 (bttv)
  - Aimslab, Aztech, ADS, Miro, GemTeK, and many more radio cards

All can (in theory :-) be used with just one application.

Video4Linux

Video4Linux provides:

- API

  Application Programming Interface defined in `videodev.h`

- Videodev module

  Driver registration and administration handled in `videodev.c`

Video4Linux standard page:

`http://roadrunner.swansea.uk.linux.org/v4l.shtml`

Video4Linux: Videodev

All Video4Linux drivers have to register with videodev, e.g. bttv:

```
video_register_device(&btv->video_dev,VFL_TYPE_GRABBER);
video_register_device(&btv->vbi_dev,VFL_TYPE_VBI);
video_register_device(&btv->radio_dev,VFL_TYPE_RADIO);
```

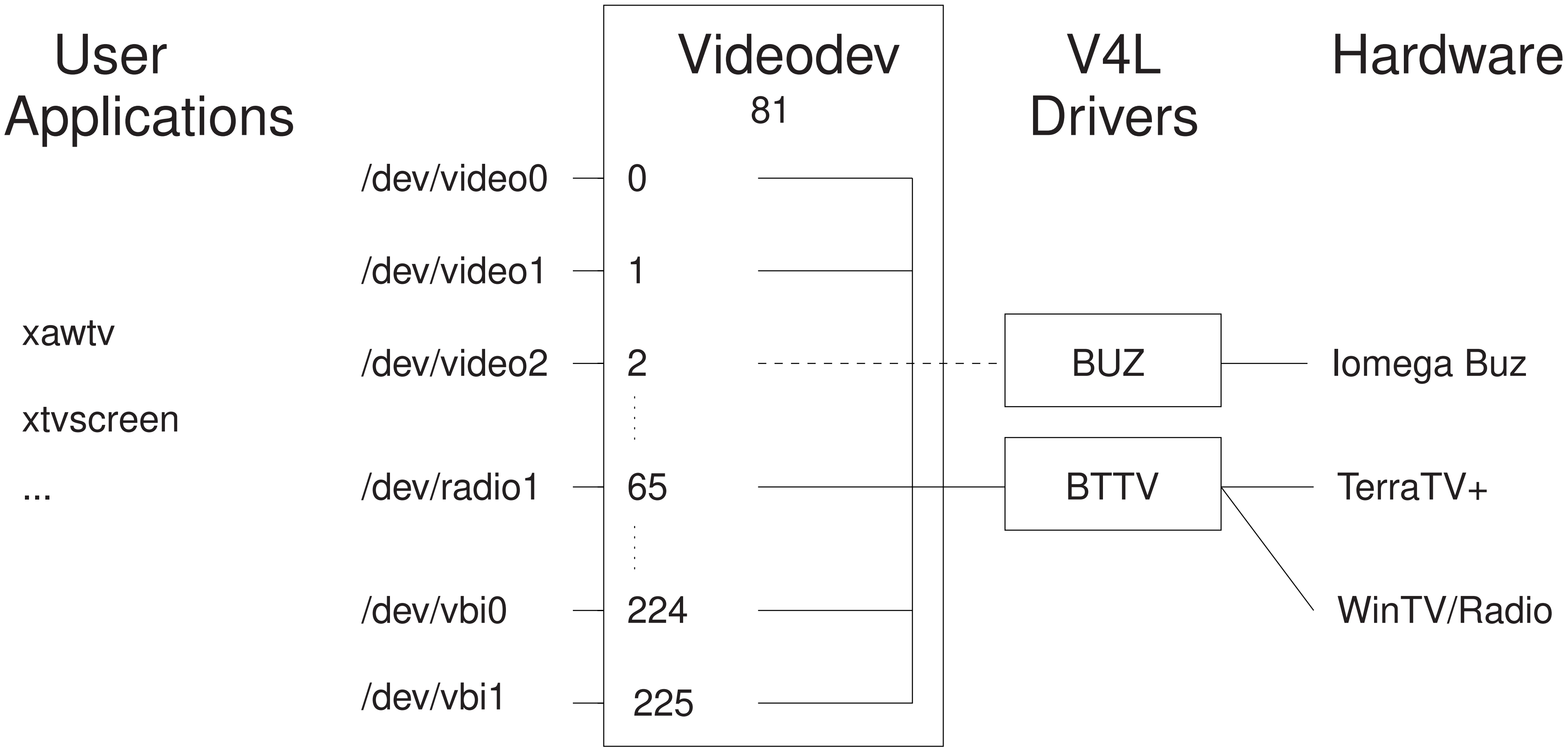
where `btv` is of type `struct bttv`:

```
struct bttv
{
  struct video_device video_dev;
  struct video_device radio_dev;
  struct video_device vbi_dev;
  ...
}
```

```
struct video_device
{
    char name[32];
    int type;
    int hardware;

    int (*open)(struct video_device *, int mode);
    void (*close)(struct video_device *);
    long (*read)(struct video_device *, char *, unsigned long, int noblock);
    long (*write)(struct video_device *, const char *, unsigned long, int noblock);
    unsigned int (*poll)(struct video_device *, struct file *, poll_table *);
    int (*ioctl)(struct video_device *, unsigned int , void *);
    int (*mmap)(struct video_device *, const char *, unsigned long);
    int (*initialize)(struct video_device *);
    void *priv;             /* Used to be 'private' but that upsets C++ */
    int busy;
    int minor;
};
```

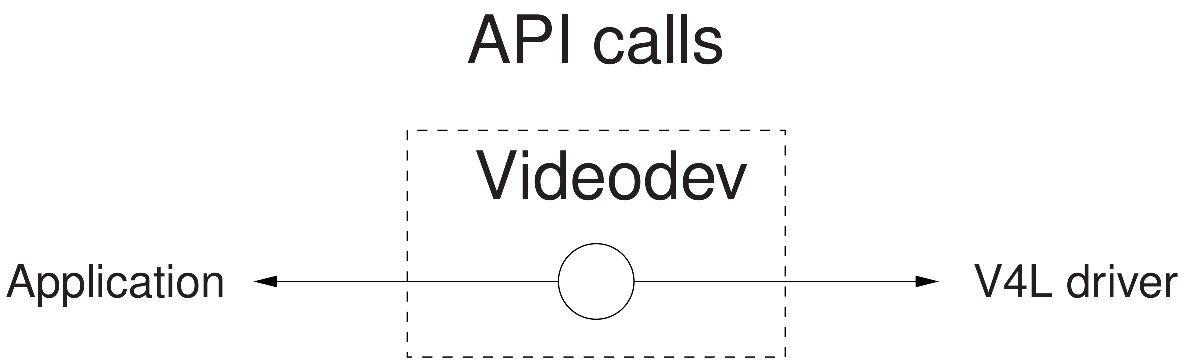
Video4Linux: Videodev



Video4Linux: API

API calls usually pass unchanged from the application to the driver.

Videodev only takes care that the right driver is reached.



But, e.g. in the case of V4L2, videodev could also do additional filtering or conversions.

Video4Linux: API

Applications should now first adapt themselves to the capabilities of the driver.

- Get capabilities: `ioctl(f, VIDIOCGCAP, &vcap);`

```
struct video_capability
{
 char name[32];
 int type;
 int channels;  /* Num channels */
 int audios;    /* Num audio devices */
 int maxwidth;  /* Supported width */
 int maxheight; /* And height */
 int minwidth;  /* Supported width */
 int minheight; /* And height */
};
```

```
#define VID_TYPE_CAPTURE        1       /* Can capture */
#define VID_TYPE_TUNER          2       /* Can tune */
#define VID_TYPE_TELETEXT       4       /* Does teletext */
#define VID_TYPE_OVERLAY        8       /* Overlay onto frame buffer */
#define VID_TYPE_CHROMAKEY      16      /* Overlay by chromakey */
#define VID_TYPE_CLIPPING       32      /* Can clip */
#define VID_TYPE_FRAMERAM       64      /* Uses the frame buffer memory */
#define VID_TYPE_SCALES         128     /* Scalable */
#define VID_TYPE_MONOCHROME     256     /* Monochrome only */
#define VID_TYPE_SUBCAPTURE     512     /* Can capture subareas of the image */
```

Video4Linux: API

- Get channel: `ioctl(f, VIDIOCGCHAN, &vchan);`

```
struct video_channel
{
  int channel;
  char name[32];
  int tuners;
  __u32  flags;
#define VIDEO_VC_TUNER  1
#define VIDEO_VC_AUDIO  2
  __u16  type;
#define VIDEO_TYPE_TV     1
#define VIDEO_TYPE_CAMERA 2
  __u16 norm;
};
```

- `VIDIOCSPICT` - Set picture properties
- `VIDIOCSFREQ` - Set tuner
- `VIDIOCSAUDIO` - Audio source, mute etc
- `VIDIOCSYNC`, `VIDIOCMCAPTURE` - Sync to and capture frames
- `VIDIOCSFBUF` - Set frame buffer
- `...`

XFree 4.0

With the Xv extension X11 can talk directly to the V4L driver:

- Video output is logically connected to a window *inside* X11 not just through the application
- Frame buffer address is passed from X11 to the driver

  (no additional root priviliges needed)
- Video is synchronized to X11 (no artifacts, no freezing)
- Direct access to clipping information

Preliminary support for Xv can be found in xawtv 3.0.

Video4Linux: I2C

I2C subsystems on multimedia cards:

- TV card
  - TV decoder (+ encoder)
  - tuner (Philips, TEMIC)
  - stereo decoder (MSP34XX)
  - teletext decoder
- DVB card
  - tuner
  - QPSK demodulator
  - AC3 decoder
  - CSS copy protection module

Video4Linux: I2C

Currently in use: I2C modules by Gerd Knorr

```
memcpy(&(btv->i2c),&bttv_i2c_bus_template,sizeof(struct i2c_bus));
btv->i2c.data = btv;
i2c_register_bus(&btv->i2c);

static struct i2c_bus bttv_i2c_bus_template =
{
        "bt848",
        I2C_BUSID_BT848,
        NULL,
        SPIN_LOCK_UNLOCKED,
        attach_inform,
        detach_inform,
        i2c_setlines,
        i2c_getdataline,
        I2CRead,
        I2CWrite,
};
```

I2C device modules can register and then access the bus:

```
i2c_register_driver(&i2c_driver_tuner);

...

i2c_write(t->bus, t->addr, tun->config, config, 1);
```

```
struct i2c_driver i2c_driver_tuner =
{
  "tuner",                        /* name       */
  I2C_DRIVERID_TUNER,             /* ID         */
  0xc0, 0xce,                     /* addr range */
  tuner_attach,
  tuner_detach,
  tuner_command
};
```

Video4Linux: I2C

Another I2C interface was written by Simon Vogel (http://www.tk.uni-linz.ac.at/˜simon/private/i2c/) or now lm_sensors package (`http://www.netroedge.com/~lm78`).

It provides drivers and applications for sensor chips temperature and fan speed sensors

- LM 75/78/80, ADM9240, Maxilife, ...

on various main boards

- Ali15x3, PIIX4, VIA, ...

and support for video cards and TV encoders (only Voodoo3 and Bt869 for now).

Video4Linux2

Video4Linux lacked several features:

- Flexible buffer sizes and number
- Compressed video input
- Video output support

In 1998 Bill Dirks introduced the new Video4Linux2 standard. Additionally to the above mentioned it features:

- Standard library
- V4L compatibility
- Standard image formats
- VBI standard specification

Video4Linux2

- API and example drivers:

  `http://millennium.diads.com/bdirks/v4l2.htm`

- Video4Linux2 library:

  `http://millennium.diads.com/mschimek/libv4l2/`

New features and applications

- BTTV
  - changes for new kernel versions
  - infra-red remote drivers for various cards (Hauppauge, Flyvideo, ...)

    `http://fsinfo.cs.uni-sb.de/~columbus/lirc/`
  - experimental Video4Linux2 version by Justin Schoeman

    `http://www.ee.up.ac.za/~justin/v4l2/`
  - GTK application with support for Video4Linux2 and DVB

    `http://www.thp.uni-koeln.de/~rjkm/linux/bttv.html`

New features and applications

- RTjpeg by Justin Schoeman

  `http://www.ee.up.ac.za/~justin/`

- Real time MPEG 1 encoder for V4L 1/2 by Michael Schimek

  `http://millennium.diads.com/mschimek/mpeg/`

MJPEG

- MJPEG

  Drivers for cards using the ZR36060:

  – Iomega Buz driver by Dave Perks

    Gernot Ziegler: `http://www.lysator.liu.se/~gz/buz/`

    Rainer Johanni: `http://www.munich-vision.de/buz/`

    including software to capture video in *movtar* format

  – LML33: `http://www.linuxmedialabs.com/`

MPEG2, DVD, DVB

DVD VOB-files: MPEG2 program streams with AC3 audio streams

DVB: MPEG2 transport streams with MPEG 1/2 audio

- Available MPEG2 card driver:
  - Stradis MPEG2 decoder (`http://mpeg.openprojects.net/`)
  - Convergence MPEG2 card (`http://www.linuxtv.org/`)

- Work is being done on
  - Matrox DVD add-on card `http://pulsar.doobie.org/`, `http://livid.on.openprojects.net/`

MPEG2, DVD, DVB

Major problems with DVD:

- No specs for MPEG2 cards
- Copy protection

  Partially solved: `http://livid.on.openprojects.net/`
- AC3 sound

  Software decoder:

Especially the *legal* situtation is still unclear.

DVB

Digital Video Broadcasting:

MPEG 2 transport stream with MPEG2 video and MPEG 1/2 audio.

- DVB cards on the market
  - Hauppauge DVB-s (`http://www.hauppauge.de/`)

    drivers under development by Convergence (`http://www.convergence.de/`)
  - KissNordic/Dooin Skyvision 1/2 (`http://www.kissnordic.dk`)

Skyvision

The Skyvision 1 card is manufactured by Dooin Electronics, Korea (`www://www.dooin.co.kr`).

The European distributor is KiSSNordic, Denmark (`www.//www.kissnordik.dk/`).

Hardware components:

- PLX9050 PCI bus bridge
- Oak OTI-8211 MPEG 2 decoder
- Zoran ZR38521 AC3 decoder
- SAA7167 RGB and STV0117 TV encoder

- Oak OTI-8511 QPSK demodulator
- Philips TSA5055 2.5 GHz tuner
- Philips PCF8574+LNBP16SP for LNB control

Skyvision Driver

- DVB
  - + channel tuning
  - + Diseqc (only tested with 4-way switch)
  - + sub-stream extraction
  - - automatic channel search, EPG, Teletext (but working on it)
- MPEG
  - + MPEG 2 with MPEG/AC3 sound (e.g. unencrypted VOBs)
  - - MPEG 1/VCD (no idea :-( )
  - - DVD (soon!)

Binary test version will be available at:

`http://www.thp.uni-koeln.de/~rjkm/linux/skyvis.html`