

Idea behind the dvb-usb-framework

Note

1. This documentation is outdated. Please check at the DVB wiki at <https://linuxtv.org/wiki> for more updated info.
2. **deprecated:** Newer DVB USB drivers should use the dvb-usb-v2 framework.

In March 2005 I got the new Twinhan USB2.0 DVB-T device. They provided specs and a firmware.

Quite keen I wanted to put the driver (with some quirks of course) into dibusb. After reading some specs and doing some USB snooping, it realized, that the dibusb-driver would be a complete mess afterwards. So I decided to do it in a different way: With the help of a dvb-usb-framework.

The framework provides generic functions (mostly kernel API calls), such as:

- Transport Stream URB handling in conjunction with dvb-demux-feed-control (bulk and isoc are supported)
- registering the device for the DVB-API
- registering an I2C-adaptor if applicable
- remote-control/input-device handling
- firmware requesting and loading (currently just for the Cypress USB controllers)
- other functions/methods which can be shared by several drivers (such as functions for bulk-control-commands)
- TODO: a I2C-chunker. It creates device-specific chunks of register-accesses depending on length of a register and the number of values that can be multi-written and multi-read.

The source code of the particular DVB USB devices does just the communication with the device via the bus. The connection between the DVB-API-functionality is done via callbacks, assigned in a static device-description (struct dvb_usb_device) each device-driver has to have.

For an example have a look in drivers/media/usb/dvb-usb/vp7045*.

Objective is to migrate all the usb-devices (dibusb, cinergyT2, maybe the ttusb; flexcop-usb already benefits from the generic flexcop-device) to use the dvb-usb-lib.

TODO: dynamic enabling and disabling of the pid-filter in regard to number of feeds requested.

Supported devices

See the LinuxTV DVB Wiki at <https://linuxtv.org> for a complete list of cards/drivers/firmwares:
https://linuxtv.org/wiki/index.php/DVB_USB

0. History & News:

2005-06-30

- added support for WideView WT-220U (Thanks to Steve Chang)

2005-05-30

- added basic isochronous support to the dvb-usb-framework
- added support for Conexant Hybrid reference design and Nebula DigiTV USB

2005-04-17

- all dibusb devices ported to make use of the dvb-usb-framework

2005-04-02

- re-enabled and improved remote control code.

2005-03-31

- ported the Yakumo/Hama/Typhoon DVB-T USB2.0 device to dvb-usb.

2005-03-30

- first commit of the dvb-usb-module based on the dibusb-source. First device is a new driver for the TwinhanDTV Alpha / MagicBox II USB2.0-only DVB-T device.
- (change from dvb-dibusb to dvb-usb)

2005-03-28

- added support for the AVerMedia AverTV DVB-T USB2.0 device (Thanks to Glen Harris and Jiun-Kuei Jung AVerMedia)

2005-03-14

- added support for the Typhoon/Yakumo/HAMA DVB-T mobile USB2.0

2005-02-11

- added support for the KWorld/ADSTech Instant DVB-T USB2.0. Thanks a lot to Joachim von Caron

2005-02-02 - added support for the Hauppauge Win-TV Nova-T USB2

2005-01-31 - distorted streaming is gone for USB1.1 devices

2005-01-13

- moved the mirrored `pid_filter` table back to `dvb-dibusb` first almost working version for HanfTek UMT-010 found out, that Yakumo/HAMA/Typhoon are predecessors of the HanfTek UMT-010

2005-01-10

- refactoring completed, now everything is very delightful
- tuner quirks for some weird devices (Artec T1 AN2235 device has sometimes a Panasonic Tuner assembled). Tunerprobing implemented. Thanks a lot to Gunnar Wittich.

2004-12-29

- after several days of struggling around bug of no returning URBs fixed.

2004-12-26

- refactored the `dibusb-driver`, split into separate files
- `i2c-probing` enabled

2004-12-06

- possibility for demod `i2c-address` probing
- new usb IDs (Compro, Artec)

2004-11-23

- merged changes from `DiB3000MC_ver2.1`
- revised the debugging
- possibility to deliver the complete TS for USB2.0

2004-11-21

- first working version of the `dib3000mc/p` frontend driver.

2004-11-12

- added additional remote control keys. Thanks to Uwe Hanke.

2004-11-07

- added remote control support. Thanks to David Matthews.

2004-11-05

- added support for a new devices (Grandtec/Avermedia/Artec)
- merged my changes (for `dib3000mb/dibusb`) to the `FE_REFACTORING`, because it became `HEAD`
- moved transfer control (`pid filter`, `fifo control`) from usb driver to frontend, it seems better settled there (added `xfer_ops-struct`)
- created a common files for frontends (`mc/p/mb`)

2004-09-28

- added support for a new device (Unknown, vendor ID is Hyper-Paltek)

2004-09-20

- added support for a new device (Compro DVB-U2000), thanks to Amaury Demol for reporting
- changed usb TS transfer method (several urbs, stopping transfer before setting a new pid)

2004-09-13

- added support for a new device (Artec T1 USB TVBOX), thanks to Christian Motschke for reporting

2004-09-05

- released the `dibusb` device and `dib3000mb-frontend` driver (old news for `vp7041.c`)

2004-07-15

- found out, by accident, that the device has a TUA6010XS for PLL

2004-07-12

- figured out, that the driver should also work with the CTS Portable (Chinese Television System)

2004-07-08

- firmware-extraction-2.422-problem solved, driver is now working properly with firmware extracted from 2.422
- #if for 2.6.4 (dvb), compile issue
- changed firmware handling, see vp7041.txt sec 1.1

2004-07-02

- some tuner modifications, v0.1, cleanups, first public

2004-06-28

- now using the dvb_dmx_swfilter_packets, everything runs fine now

2004-06-27

- able to watch and switching channels (pre-alpha)
- no section filtering yet

2004-06-06

- first TS received, but kernel oops :/

2004-05-14

- firmware loader is working

2004-05-11

- start writing the driver

How to use?

Firmware

Most of the USB drivers need to download a firmware to the device before start working.

Have a look at the Wikipage for the DVB-USB-drivers to find out, which firmware you need for your device:

https://linuxtv.org/wiki/index.php/DVB_USB

Compiling

Since the driver is in the linux kernel, activating the driver in your favorite config-environment should sufficient. I recommend to compile the driver as module. Hotplug does the rest.

If you use dvb-kernel enter the build-2.6 directory run 'make' and 'insmod.sh load' afterwards.

Loading the drivers

Hotplug is able to load the driver, when it is needed (because you plugged in the device).

If you want to enable debug output, you have to load the driver manually and from within the dvb-kernel cvs repository.

first have a look, which debug level are available:

```
System Message: WARNING/2 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\media\drivers\linux-master) (Documentation) (driver-api) (media) (drivers) dvb-usb.rst, line 259)
```

Cannot analyze code. No Pygments lexer found for "none".

```
.. code-block:: none

    # modinfo dvb-usb
    # modinfo dvb-usb-vp7045

    etc.
```

```
System Message: WARNING/2 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\media\drivers\linux-master) (Documentation) (driver-api) (media) (drivers) dvb-usb.rst, line 266)
```

Cannot analyze code. No Pygments lexer found for "none".

```
.. code-block:: none

    modprobe dvb-usb debug=<level>
    modprobe dvb-usb-vp7045 debug=<level>
```

should do the trick.

When the driver is loaded successfully, the firmware file was in the right place and the device is connected, the "Power"-LED should be turned on.

At this point you should be able to start a dvb-capable application. I'm use (t|s)zap, mplayer and dvbscan to test the basics. VDR-xine provides the long-term test scenario.

Known problems and bugs

- Don't remove the USB device while running an DVB application, your system will go crazy or die most likely.

Adding support for devices

TODO

USB1.1 Bandwidth limitation

A lot of the currently supported devices are USB1.1 and thus they have a maximum bandwidth of about 5-6 MBit/s when connected to a USB2.0 hub. This is not enough for receiving the complete transport stream of a DVB-T channel (which is about 16 MBit/s). Normally this is not a problem, if you only want to watch TV (this does not apply for HDTV), but watching a channel while recording another channel on the same frequency simply does not work very well. This applies to all USB1.1 DVB-T devices, not just the dvb-usb-devices)

The bug, where the TS is distorted by a heavy usage of the device is gone definitely. All dvb-usb-devices I was using (Twinhan, Kworld, DiBcom) are working like charm now with VDR. Sometimes I even was able to record a channel and watch another one.

Comments

Patches, comments and suggestions are very very welcome.

3. Acknowledgements

Amaury Demol (Amaury.Demol@parrot.com) and Francois Kanounnikoff from DiBcom for providing specs, code and help, on which the dvb-dibusb, dib3000mb and dib3000mc are based.

David Matthews for identifying a new device type (Artec T1 with AN2235) and for extending dibusb with remote control event handling. Thank you.

Alex Woods for frequently answering question about usb and dvb stuff, a big thank you.

Bernd Wagner for helping with huge bug reports and discussions.

Gunnar Wittich and Joachim von Caron for their trust for providing root-shells on their machines to implement support for new devices.

Allan Third and Michael Hutchinson for their help to write the Nebula digitv-driver.

Glen Harris for bringing up, that there is a new dibusb-device and Jiun-Kuei Jung from AVerMedia who kindly provided a special firmware to get the device up and running in Linux.

Jennifer Chen, Jeff and Jack from Twinhan for kindly supporting by writing the vp7045-driver.

Steve Chang from WideView for providing information for new devices and firmware files.

Michael Paxton for submitting remote control keymaps.

Some guys on the linux-dvb mailing list for encouraging me.

Peter Schildmann <peter.schildmann-nospam-at-web.de> for his user-level firmware loader, which saves a lot of time (when writing the vp7041 driver)

Ulf Hermenau for helping me out with traditional chinese.

Andr   Smoktun and Christian Fr  mmel for supporting me with hardware and listening to my problems very patiently.