c-qcam - Connectix Color QuickCam video4linux kernel driver

1999-12-08 Dave Forrest, written with kernel version 2.2.12 in mind

Table of Contents

1.0 Introduction

The file ../../drivers/media/video/c-qcam.c is a device driver for the Logitech (nee Connectix) parallel port interface color CCD camera. This is a fairly inexpensive device for capturing images. Logitech does not currently provide information for developers, but many people have engineered several solutions for non-Microsoft use of the Color Quickcam.

1.1 Motivation

I spent a number of hours trying to get my camera to work, and I hope this document saves you some time. My camera will not work with the 2.2.13 kernel as distributed, but with a few patches to the module, I was able to grab some frames. See 4.0, Future Work.

2.0 Compilation, Installation, and Configuration

The c-qcam depends on parallel port support, video4linux, and the Color Quickcam. It is also nice to have the parallel port readback support enabled. I enabled these as modules during the kernel configuration. The appropriate flags are:

```
    CONFIG_PRINTER       M    for lp.o, parport.o parport_pc.o modules
    CONFIG_PNP_PARPORT   M for autoprobe.o IEEE1284 readback module
    CONFIG_PRINTER_READBACK M for parport_probe.o IEEE1284 readback module
    CONFIG_VIDEO_DEV     M    for videodev.o video4linux module
    CONFIG_VIDEO_CQCAM   M    for c-qcam.o  Color Quickcam module
```

With these flags, the kernel should compile and install the modules. To record and monitor the compilation, I use:

```
 (make zlilo ; \
  make modules; \
  make modules_install ;
  depmod -a ) &>log &
 less log  # then a capital 'F' to watch the progress
```

But that is my personal preference.

2.2 Configuration

The configuration requires module configuration and device configuration. I like kmod or kerneld process with the /etc/modprobe.conf file so the modules can automatically load/unload as they are used. The video devices could already exist, be generated using MAKEDEV, or need to be created. The following sections detail these procedures.

2.1 Module Configuration

Using modules requires a bit of work to install and pass the parameters. Understand that entries in /etc/modprobe.conf of:

```
   alias parport_lowlevel parport_pc
   options parport_pc io=0x378 irq=none
   alias char-major-81 videodev
   alias char-major-81-0 c-qcam
```

will cause the kmod/modprobe to do certain things. If you are using kmod, then a request for a 'char-major-81-0' will cause the 'c-qcam' module to load. If you have other video sources with modules, you might want to assign the different minor numbers to different modules.

2.2 Device Configuration

At this point, we need to ensure that the device files exist. Video4linux used the /dev/video* files, and we want to attach the Quickcam to one of these.

```
   ls -lad /dev/video*  # should produce a list of the video devices
```

If the video devices do not exist, you can create them with:

```
  su
  cd /dev
  for ii in 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 ; do
    mknod video$ii c 81 $ii   # char-major-81-[0-16]
    chown root.root video$ii  # owned by root
    chmod 600 video$ii        # read/writable by root only
  done
```

Lots of people connect video0 to video and bttv, but you might want your c-qcam to mean something more:

```
   ln -s video0 c-qcam  # make /dev/c-qcam a working file
   ln -s c-qcam video   # make /dev/c-qcam your default video source
```

But these are conveniences. The important part is to make the proper special character files with the right major and minor numbers. All of the special device files are listed in ../devices.txt. If you would like the c-qcam readable by non-root users, you will need to change the permissions.

3.0 Troubleshooting

If the sample program below, v4lgrab, gives you output then everything is working.

```
    v4lgrab | wc # should give you a count of characters
```

Otherwise, you have some problem.

The c-qcam is IEEE1284 compatible, so if you are using the proc file system (CONFIG_PROC_FS), the parallel printer support (CONFIG_PRINTER), the IEEE 1284 system,(CONFIG_PRINTER_READBACK), you should be able to read some identification from your quickcam with

```
         modprobe -v parport
         modprobe -v parport_probe
         cat /proc/parport/PORTNUMBER/autoprobe
Returns:
  CLASS:MEDIA;
  MODEL:Color QuickCam 2.0;
  MANUFACTURER:Connectix;
```

A good response to this indicates that your color quickcam is alive and well. A common problem is that the current driver does not reliably detect a c-qcam, even though one is attached. In this case,

```
     modprobe -v c-qcam
or
     insmod -v c-qcam
```

Returns a message saying "Device or resource busy" Development is currently underway, but a workaround is to patch the module to skip the detection code and attach to a defined port. Check the video4linux mailing list and archive for more current information.

3.1 Checklist:

Can you get an image?

```
            v4lgrab >qcam.ppm ; wc qcam.ppm ; xv qcam.ppm
```

Is a working c-qcam connected to the port?

```
            grep ^ /proc/parport/?/autoprobe
```

Do the /dev/video* files exist?

```
            ls -lad /dev/video
```

Is the c-qcam module loaded?

```
            modprobe -v c-qcam ; lsmod
```

Does the camera work with alternate programs? cqcam, etc?

4.0 Future Work / current workarounds

It is hoped that this section will soon become obsolete, but if it isn't, you might try patching the c-qcam module to add a parport=xxx option as in the bw-qcam module so you can specify the parallel port:

```
       insmod -v c-qcam parport=0
```

And bypass the detection code, see ../../drivers/char/c-qcam.c and look for the 'qc_detect' code and call.

Note that there is work in progress to change the video4linux API, this work is documented at the video4linux2 site listed below.

9.0 --- A sample program using v4lgrabber,

v4lgrab is a simple image grabber that will copy a frame from the first video device, /dev/video0 to standard output in portable pixmap format (.ppm) To produce .jpg output, you can use it like this: 'v4lgrab | convert - c-qcam.jpg'

10.0 --- Other Information

Use the ../../Maintainers file, particularly the VIDEO FOR LINUX and PARALLEL PORT SUPPORT sections

The video4linux page:
  http://linuxtv.org

The V4L2 API spec:
  http://v4l2spec.bytesex.org/

Some web pages about the quickcams:
   http://www.dkfz-heidelberg.de/Macromol/wedemann/mini-HOWTO-cqcam.html

http://www.crynwr.com/qcpc/	QuickCam Third-Party Drivers
http://www.crynwr.com/qcpc/re.html	Some Reverse Engineering
http://cse.unl.edu/~cluening/gqcam/	v4l client
http://phobos.illtel.denver.co.us/pub/qcread/	doesn't use v4l
ftp://ftp.cs.unm.edu/pub/chris/quickcam/	Has lots of drivers
http://www.cs.duke.edu/~reynolds/quickcam/	Has lots of information