Video Editing on Ubuntu 8.04 with Cinelerra, outputting to avi file

Introduction:

This guide is for Ubuntu 8.04 users who want to edit video in Cinelerra and export it to an MPEG4 avi file. I have used PAL settings for export as I live in the UK. This guide only concerns the setup process, so users wishing to learn how to use Cinelerra should refer to the documentation on the website (see link below).

All I can say is that this worked for me. There are no guarantees this will work for you.

For reference, I am using a dual core AMD 2.8GHz machine with 4GB of RAM and running Ubuntu 8.04 64-bit.

The guide will involve adding a new repository to download Cinelerra and build FFMPEG from source if you want to convert files to avi. It will also cover the basic syntax required to convert files to another format using FFMPEG.

Part 1 - Getting Cinelerra:

Get Cinelerra from here using the instructions provided:

http://cvs.cinelerra.org/getting_cinelerra.php#ubuntu

```
sudo wget http://akirad.cinelerra.org/dists/hardy.list -O /etc/apt/sources.list.d/akirad.list

wget -q http://akirad.cinelerra.org/dists/akirad.key -O- | sudo apt-key add - && sudo apt-get update

sudo apt-get install cinelerra-generic
```

Cinelerra must be set to work with PulseAudio. Open Cinelerra and go to Settings → Preferences → Playback → Audio Driver. Select ESound and set the following parameters:

```
Server: [leave blank]
Port: 7007
```

Then downloaded the manual from here: http://cvs.cinelerra.org/docs/cinelerra_cv_manual_en.pdf

Main documentation page here: http://cvs.cinelerra.org/docs.php

Part 2 - Outputting a file from Cinelerra:

Imported an avi video and made changes. Saved the project – very important if anything goes wrong.
Then following manual suggestions for exporting video making modifications for PAL video standard (section entitled *20.9.1.2 yuv4mpeg pipe through ffmpeg*):

- Select File -> Render... or press SHIFT-R. The render dialogue pops up.
- In the render dialogue, you have the choice to render 1. the entire project, or 2. the highlighted selection, or 3. from In-point "[" to Out-point "]".
- Make sure the Insertion strategy is "Create new resources only".
- Select the AC3 audio output file format.
- Specify the audio output file name and path (example: input.ac3).
- Select Render audio tracks and deselect Render video tracks.
- Click on the wrench next to "Audio:". A new dialogue "Cinelerra: Audio Compression" pops up.
- Set the bitrate to 192 kbps (or whatever).
- Click OK, the compression dialogue disappears.
- In the render dialogue, click OK, the dialogue disappears. Audio is rendered. Rendering audio is much faster than rendering video but might still take some seconds. Watch the progress bar in the main window’s lower right corner.
- Again, press SHIFT-R. The render dialogue pops up again.
- Select the YUV4MPEG Stream file format.
- Specify the video output file name and path (example: input.m2v).
- Deselect Render audio tracks and select Render video tracks.

- Click on the wrench next to "Video:". A new dialogue window "Cinelerra: YUV4MPEG stream" pops up. The first text box should already contain the output file name and path you had specified in the render dialogue.
- Select "Use Pipe:".
- Fill the following command line into the second text box:

```
  ffmpeg -f yuv4mpegpipe -i - -y -target pal-dvd -flags +ilme+ildct %
```

- Click OK in the yuv4mpeg dialogue and in the render dialogue to render video output.
- The resulting .m2v can be further processed together with the .ac3 audio with the following shell command, producing a dvd-compatible mpeg stream:

```
  ffmpeg -i input.ac3 -i input.m2v -target pal-dvd -flags +ilme+ildct output.mpg
```

(Yes, the stream is sent through ffmpeg a second time.)

At this point I discovered that FFMPEG in Ubuntu 8.04 is not compiled with support for AC3 or MP3, so again followed instructions to build from source (with modifications) referring first to page here: https://wiki.ubuntu.com/ffmpeg

What follows is the method I used to install FFMPEG with all the codecs I could ever want installed!

Note: Ensure you change the version number as described below or the automatic package updater may very well update the version and remove all the custom configuration that allows mp3 and ac3 to be used.

Initially, I had problems with libx264 compilation as when running `make` it caused the `error: libavcodec/libx264.c:224: error: ‘X264_ME_TESA’ undeclared (first use in this function).` Also had problems when running FFMPEG when it tried to find libswscale.

Get FFMPEG:
http://ffmpeg.mplayerhq.hu/download.html

Do:

```
svn checkout svn://svn.mplayerhq.hu/ffmpeg/trunk ffmpeg
```

Get X264 snapshot:
ftp://ftp.videolan.org/pub/videolan/x264/snapshots/

Build X264 first using following instructions:

```
./configure --enable-shared --disable-asm
make
sudo install
```

The instructions used for building FFMPEG I found here (with slight modifications):

http://stream0.org/2008/01/install-ffmpeg-on-ubuntu-gutsy.html

```
sudo apt-get build-dep ffmpeg

cd ffmpeg

sudo apt-get install liblame-dev libfaad-dev libfaad2-dev libfaac-dev libxvidcore4-dev liba52-0.7.4
liba52-0.7.4-dev libdts-dev checkinstall libx264-dev checkinstall build-essential subversion

./configure --enable-gpl --enable-pp --enable-libvorbis --enable-libtheora --enable-liba52 --enable-
libdc1394 --enable-libgsm --enable-libmp3lame --enable-libfaad --enable-libfaac --enable-libxvid
--enable-pthreads --enable-libx264 --enable-shared

make

sudo checkinstall
```

We can now change some of the package settings.
When presented with the series of options, press enter all the way through to accept the defaults, or change them as specified below to prevent Ubuntu automatically prompting you to install what it believes to be a more up-to-date version of FFMPEG.

Simply enter the number of the option you want to change and press enter.

- Changing option 0 will allow you to put your own name as the package maintainer.
- Changing option 3 will allow you to change the version number. You can alter this as I have done to change from `0.cvs20070307` to `4:0svn20080707` – **this should prevent package manager from trying to overwrite your setup with a new package.**

See below for an example of what the output would look like.

```
This package will be built according to these values:
0 -  Maintainer: [ Rob Whalley mail@robwhalley.co.uk ]
1 -  Summary: [ Package created with checkinstall 1.6.1 ]
2 -  Name:    [ ffmpeg ]
3 -  Version: [ 4:0.svn20080707 ]
4 -  Release: [ 1 ]
5 -  License: [ GPL ]
6 -  Group:   [ checkinstall ]
7 -  Architecture: [ amd64 ]
8 -  Source location: [ 4:0svn20080707 ]
9 -  Alternate source location: [  ]
10 - Requires: [  ]

Enter a number to change any of them or press ENTER to continue:
```

If successful, you should get a message similar to:

```
Done. The new package has been installed and saved to
/home/rob/Compile/ffmpeg/ffmpeg_4:0.svn20080707-1_amd64.deb
You can remove it from your system anytime using:
dpkg -r ffmpeg
```

If for whatever reason you have problems, refer to the original documentation link above.

You can now use the following FFMPEG command to export the sound file (input.ac3) and the video file (input.m2v) to an MPG file:

ffmpeg -i input.ac3 -i input.m2v -target pal-dvd -flags +ilme+ildct output.mpg

Make sure you change to the folder you have stored the files in first though (e.g. cd /home/rob/Videos)
This will give you an mpg file, but what if you want an avi file?... You can try:

ffmpeg -i input.ac3 -i input.m2v -f avi -vcodec mpeg4 -b 666k -g 300 -bf 2 -acodec libmp3lame -ab 192k output.avi

the -f switch forces the avi file format, -ab is the audio bitrate (in my case 192kbps), and -b is the video bitrate, which I have calculated as 666kbps using the following formulae:

*bitrate = (target_size_in_Mbytes - sound_size_in_Mbytes) * 1024 * 1024 / length_in_secs * 8 / 1000*
*bitrate = (175 - 32) * 1024 * 1024 / 1800 * 8 / 1000 = 666.428302222*

from: http://www.mplayerhq.hu/DOCS/HTML/en/menc-feat-dvd-mpeg4.html

To find out more about FFMPEG syntax, refer to the documentation here: http://ffmpeg.mplayerhq.hu/ffmpeg-doc.html

Note that it is also likely that the mencoder program (part of mplayer) can also do the encoding to AVI for you. Visit the following pages if you wish to find out more:

http://www.mplayerhq.hu/DOCS/HTML/en/mencoder.html

http://www.mplayerhq.hu/DOCS/HTML/en/encoding-guide.html

Additional Information:

From:

http://www.openmovieeditor.org/board/viewtopic.php?pid=1729

If you get error:

```
error while loading shared libraries: libswscale.so.0: cannot open shared object file: No such file or directory
```

Try:

```
export LD_LIBRARY_PATH=/usr/local/lib
```

Tests done:

Export using quality 80000:
Approx 320MB / 3 mins,

Export using 8000:
Approx 125MB / 3 mins

Then:

```
ffmpeg -i test.avi -f avi -vcodec mpeg4 -b 1600k -g 300 -bf 2 -acodec libmp3lame -ab 192k output1.avi
```

File size: 27.6MB

or mplayer using ffmpeg (2 pass, slower):

```
mencoder test.avi -oac mp3lame -ovc xvid -xvidencopts pass=1 -o /dev/null
mencoder test.avi -oac mp3lame -ovc xvid -xvidencopts pass=2:bitrate=1600 -o output2.avi
```

File size: 25.0MB

Quality of both comparable, difficult to tell difference.

http://www.mplayerhq.hu/DOCS/HTML/en/menc-feat-mpeg4.html

```
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:vpass=1 -oac mp3lame -lameopts vbr=3 -o /dev/null
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell:vpass=2 -oac mp3lame -lameopts vbr=3 -o output3.avi
```

Definite quality loss but not too bad, resulting file size: 14.4MB

As above but with video and audio bitrate specified:

```
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:vbitrate=1600:vpass=1 -oac mp3lame -lameopts abr=192 -o /dev/null
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:vbitrate=1600:mbd=2:trell:vpass=2 -oac mp3lame -lameopts abr=192 -o output4.avi
```

Output size: 26.8MB, similar quality to original two tests.

At half the VBR:

```
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:vbitrate=800:vpass=1 -oac mp3lame -lameopts abr=192 -o /dev/null
mencoder test.avi -ovc lavc -lavcopts vcodec=mpeg4:vbitrate=800:mbd=2:trell:vpass=2 -oac mp3lame -lameopts abr=192 -o output5.avi
```

Output size: 14.4MB, obviously quality reduced but not a bad bet for lower bitrate stuff for the Internet.