Guide to DVD Chapter 8 Examples:
Gabriel Maldonado

Working with Control Signals

Libraries

These examples use four libraries, two of which you can build from the sources supplied here, *portsf* and *tiny*. The third and fourth, *portaudio* and *fltk*, can be downloaded and installed from its main distribution site.

portsf

To build this library, go to the portsf sub-directory and run make:

```
$ make
```

But if you are on OSX 10.6, you will need to add some options to make (this is because a required library FLTK does not build for the default 10.6 architecture. x86_64, so all code used needs to be built for i386):

```
$ make CC="gcc -arch i386"
```

To install it ready for your own use, just type:

```
$ make install
```

tiny

To build this library, go to the TinyAudioLibrary sub-directory and run make:

```
$ make
```

But if you are on OSX 10.6, you will need to add some options to make (as explained above):

```
$ make CC="gcc -arch i386"
```

To install it ready for your own use, just type:

```
$ make install
```

portaudio

Instructions for downloading and installing Portaudio (v.19) this are found in http://portmedia.sourceforge.net/

fltk

FLTK can be downloaded from http://www.fltk.org . In this examples we are using version 1.1.* (e.g. 1.1.9), of which you can download the latest stable sources. After extracting the archive files, change to the sources directory and type:

```
$ ./configure --enable-shared –enable-threads
```

After this configuration step, you can build with:

```
$ make
```

But if you are on OSX 10.6, FLTK will not build for the default architecture, so you will need to add some options to make:

```
$ make CC="gcc -arch i386" CXX="g++ -arch i386"
```

To install it in your system, just type (you will need the admin password for this):

```
$ sudo make install
```

Now you have fltk installed (in /usr/local) and can proceed to build the examples.

LFO examples

These examples, from *Working with Control Signals*, as above, only require portaudio and fltk. The lfo1-6 programs can be built with the following command:

```
g++ -o lfo1 LFO1gui.cpp LFO1.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
      -lfltk –lportaudio
```

If you are on OSX10.6, remember to add –arch i386 to the g++ command:

```
$ g++ -arch i386 ...
```

[NB: this will be required all the other examples to be successfully built on OSX10.6]

```
g++ -o lfo2 LFO2gui.cpp LFO2.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
      -lfltk -lportaudio

g++ -o lfo3 LFO3gui.cpp LFO3.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
      -lfltk -lportaudio

g++ -o lfo4 LFO4gui.cpp LFO4.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
      -lfltk -lportaudio

g++ -o lfo5 LFO5gui.cpp LFO5.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
      -lfltk -lportaudio

g++ -o lfo6 LFO6gui.cpp LFO6.cpp PortAudioGUI.cpp PortAudioStuff.cpp
      -I./include -I/usr/local/include -L./lib -L/usr/local/lib
```

```
-lfltk –lportaudio
```

and can be run, as above with:

```
$ ./lfo1
```

etc…. Enjoy!