# A little light music
# STUDIO TIME



**Making music with Linux is now becoming easier. Soon we could all be the next number one. Jack Owen looks at the possibilities of MIDI on Linux**

Bored with the usual chart fodder, we decided to make a fortune with Linux by producing the next big song at home. Having listened to MTV we feel we could do better. To make our masterpiece we could arrange for an ensemble of session musicians to perform the music while we record it. Getting everyone to play perfectly at the same time can be frustrating, so it is usually better to try and record individuals and then layer these tracks on top of one another. This allows finer control of variables such as volume and the ability to cut and paste sections. However we want to avoid the expensive studio costs and session musicians and are going to produce the sounds ourselves electronically.

We could start by using editors to rearrange .wav files, but this would be very tedious. Better still would be to generate our own sounds with an attached music keyboard.

A synthesiser is an electronic device designed to produce synthetic sounds. We could use frequency modulation to generate sound waves, as this is used for some electronic music. A more popular method is to use sounds of real instruments that have been recorded. The use of these samples is referred to as wave table synthesis. Typically we may use a music
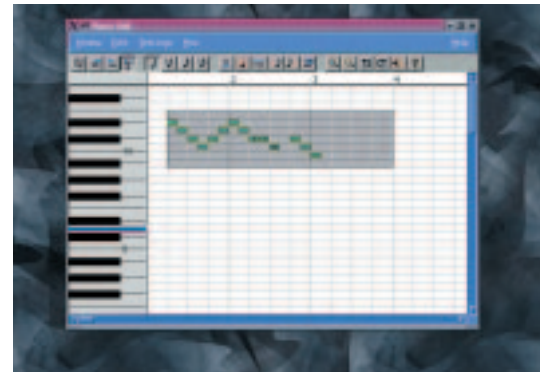

**Figure 2: PianoWin**


**Figure 3: Random noise**

keyboard to act as an input device. The keyboard acts as a controller to the sound generating hardware (your computer). Together the keyboard and the sound generator make up a synthesiser.

We are not limited to just using keyboards, however. With the MIDI protocol we can connect many devices together to generate the sound output. However we can go one stage further, as the computer can act as a sequencer.

The sequencer enables you to take input (from the keyboard or program file) data and rearrange it in whatever order you choose. It is usually capable of editing, rearranging and storing the data. It can then send the data as a finished arrangement to the sound generator to play.

MIDI is a data communications protocol. This
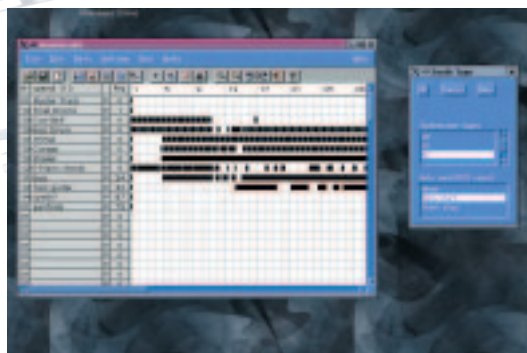

**Figure 1: The Jazz++ sequencer**

**Figure 4: Streaming Ogg Vorbis encoded BBC Radio 1**

defines the rules by which electronic musical instruments communicate. Examples of these musical instruments (devices) are synthesisers, keyboards, effect processors, recording machines and sequencers. The MIDI specification defines the format of the signals flowing from one device to the other. Such signals are commonly referred to as MIDI messages, each carrying a MIDI event. The information of an event would for example be "play note A at velocity 75" or "stop note A" or "change sound to acoustic piano".

First we need to generate some music samples. We can connect a music keyboard to a Linux box with a standard MPU-401 port (this is the joystick port on your soundcard).

By using a program such as Jazz++ (See Figure 1) we can record whatever we play on the keyboard. If you do not own a music keyboard, or do not consider yourself an accomplished player, all is not lost. You can enter notes one by one with a note editor such as pianowin (See Figure 2). This enables us to choose the note and its start and stop periods. Although this may seem a long process we only have to enter a few samples as later we will reuse and repeat the samples to build up the music track. If this is too much of a chore you could always opt for the randomly generated rhythm feature.

At this stage we have a file that represents a sound track in the computer. Once we have made up our track we can change to the sequencer section where we can start to lay down the tracks in whatever order we choose. We can modify the sound sample so the output sounds however we want it to from a high flute sound to a series of screams. We can now repeat the process and produce as many of these tracks as we want.

Using the editor feature we can then layer these



**Figure 6: Rosegarden scripting**

on top of one another. The resulting output file will now be our masterpiece. An example of the type of output that can be achieved with Jazz++ can be found on the CD in the music directory.

Now we have made our tune we really need to let everyone know about it. Being good Open Source people we obviously want to release it as an Ogg Vorbis file. Ogg Vorbis files are similar to those of MP3 file types except they do not suffer the copyright restrictions. Until the end of the year the BBC are streaming Radio 1 and occasionally Radio 4 as Ogg Vorbis files. The streamed sound is not CD quality but is excellent for radio. You will however save your own track with a higher sample rate so it is CD quality.

To convert your tune to Ogg Vorbis you need to use the Oggenc tool from the Vorbis tools package. The command to use the encoding codec is:

```
oggenc farbettr.wav
```

To play this use either the Ogg codec built into xmms or the command:

```
ogg123 farbettr.ogg
```

This now brings us to publishing free music. The Electronic Frontier Foundation has published an Open Audio License and examples of music released under this license can be found at the Open Music Registry.

Another free license system is the OpenMusic system. This produces two licenses depending on the level of use required. Sadly the Web site is currently down but on the coverdisc we have included the track Penguin Planet by Void Main. The author Dennis Gustafsson is a strong supporter of GNU/Linux and you can visit his web site at *http://mp3.com/voidmain*

So now we have produced a top ten hit, given it away and everyone is talking about it. A local music society would like to perform it and ask for the score. Again you could write it out by hand or let Linux come to the rescue. With plain text input we can use the music typesetter Lilypond. A more recent program is Rosegarden, which also features a MIDI sequencer.



**Figure 5: Output from Lilypond**

## Info

*http://www.gnu.org/software/lilypond/*
*http://www.all-day-breakfast.com/rosegarden/*
*http://www.openmusicregistry.org/*
*http://www.eff.org/IP/Open_licenses/eff_oal.html*
*http://www.jazzware.com*