Aldrin User Manual

Leonard Ritter

Aldrin User Manual
by Leonard Ritter

This book describes usage and configuration of Aldrin, a powerful music composition software for the GNU/Linux operating system.

Table of Contents

List of Figures

Chapter 1. Introduction

Welcome to the Aldrin User Manual. In this book you are going to learn how to master Aldrin, a powerful music composition software for the GNU/Linux operating system. Its compact and integrated structure allows you to write electronic music without requiring expensive audio gear, but in case you have those already, they can be easily utilized as well.

Aldrin features a flexible audio routing system commonly found in expensive audio software, enabling you to mix, split, mutilate and modify audio signals emitted by software synthesizers and samples. An ever-expanding number of included plugins ensures that you never run out of good sounds. The wavetable organizes your samples and sampler patches, enabling multiple instruments to share the same set of sounds. The classic tracker interface supports sequencing rhythms and melody patterns using your PC keyboard as a piano. Finally, a multitrack sequencer organizes your overall song structure, allowing you to arrange patterns along a vertical timeline.

With Aldrin, you're no longer alone. Exchange songs with others, remix and extend unfinished works. Aldrin embeds easy access to Freesound, the online sample library for creative commons licensed sounds. Aldrins CCM song format contains everything required to faithfully reproduce a composition on a different installation: patterns, sequences, sounds, effects, instruments, everything is preserved in a compact and extensible format.

1.1. Features

Here is a list of some of Aldrins features to wet your appetite. It can never be complete, as Aldrin is constantly being extended and improved.

- Utilizes the powerful libzzub player library for best performance and feature richness.
- Playing-piano-like pattern editor with extensive keyboard command support.
- Step sequencer with unlimited number of tracks per plugin.
- Modular router supporting generator and effect plugins.
- Plugin parameter editor with preset management and sliders to control individual parameters.
- Sound library supporting import and export of FLAC, WAV, AIF and MP3 files.
- Embedded interface to The Freesound Project. Allows to preview and download samples on the fly.
- Recording events sent from external MIDI gear and mouse cursor.
- Envelope editor for samples with ADSR generator and load/save support.
- Support for CCM as default fileformat: XML song data, Flac samples and Lunar dsp scripts packed as zip archive.
- Features the powerful Lunar DSP scripting engine.
- Support for ALSA, Jack and OSS audio drivers.

- All effects operate on stereo 32-bit floating point buffers for maximum quality.

- Supports CD- and production-grade samplerates.

- HD Recorder supports downmixing of songs to WAV for production use.

- Support for color themes.

- Support for importing Jeskola BMX files.

Chapter 2. Walkthrough Guide

The Aldrin walkthrough should give you a quick and helpful introduction into Aldrin. By the end of reading this chapter you will know how to install Aldrin, load generators and effects, write patterns and arrange those patterns in a sequence.

2.1. Installation

Aldrin is open source and therefore its releases are source code only - no binaries. If you use a popular distribution of GNU/Linux, chances are that your local package manager already offers an Aldrin package. If not, you will need to build Aldrin yourself.

Independent of which distribution you use, building and installing Aldrin shouldn't be hard, given that you follow this guide. If you have trouble installing Aldrin, please contact your distribution maintainers and ask them to include support for Aldrin.

First make sure that you have all programs and libraries installed required to build Aldrin. Usually, all of them should be provided by your distribution of GNU/Linux. Please check with your distributions package manager documentation on how to search and install packages. You are going to need following packages:

- Python 2.4 (http://www.python.org/) or later.

- wxPython 2.6 (http://www.wxpython.org/) or later, although 2.6 is recommended.

- ctypes (http://python.net/crew/theller/ctypes/), if you are using Python 2.4. Since Python 2.5, ctypes is included.

- SCons 0.96.92 (http://www.scons.org/) or later.

- pyzzub 0.2.2 (http://trac.zeitherrschaft.org/zzub/) or later. pyzzub is a specialized component based on libzzub that your distribution is most likely not supplying. Please see the installation guide for libzzub for details.

Aldrin packages can be downloaded from the SourceForge.net project website (http://aldrin.sf.net). You can either download a release or build Aldrin from the source repository. In this guide, we are going to download a release.

1. On the project website, click on Downloads to see available releases. You will see the downloads page on which one or more releases are listed.

2. Download the sources of the most recent release. Usually, the link should be named something like Aldrin *x.x Source Release (x86, .tar.bz2)*, where x.x is the release number.

3. Unpack the downloaded sources into a new folder. You will get a new subfolder named something like *aldrin-x.x*:

   ```
   [paniq@aspera tmp]$ tar -vxf aldrin-x.x.tar.bz2
   ```

```
...
aldrin-x.x/CREDITS
aldrin-x.x/INSTALL
aldrin-x.x/ChangeLog
aldrin-x.x/LICENCE
```

4. Change into the newly created folder and run scons to see if everything is fine:

```
[paniq@aspera tmp]$ cd aldrin-x.x
[paniq@aspera aldrin-x.x]$ scons
scons: Reading SConscript files ...
scons: done reading SConscript files.
scons: Building targets ...
scons: '.' is up to date.
scons: done building targets.
```

5. Please make sure that any previous versions of Aldrin have been uninstalled from your system before proceeding with the next step.

    Now install aldrin to your system (you will need root rights for this):

```
[paniq@aspera aldrin-x.x]$ sudo scons install
Password:
scons: Reading SConscript files ...
scons: done reading SConscript files.
scons: Building targets ...
Install file: "bin/aldrin" as "/usr/local/bin/aldrin"
...
scons: done building targets.
```

    If you don't want to install aldrin systemwide, create a folder in your home directory and change your installation prefix:

```
[paniq@aspera aldrin-x.x]$ mkdir ~/usr
[paniq@aspera aldrin-x.x]$ scons install PREFIX=/home/paniq/usr
scons: Reading SConscript files ...
scons: done reading SConscript files.
scons: Building targets ...
Install file: "bin/aldrin" as "/home/paniq/usr/bin/aldrin"
...
scons: done building targets.
```

You are now ready to run Aldrin for the first time.

2.2. Running Aldrin

If you installed Aldrin systemwide, you can find Aldrin in your desktop environments start menu in the section Sound & Video.

If you installed Aldrin into your homefolder, you have to run it manually. In the previous example, we installed Aldrin to `/home/paniq/usr`, which means that the applications start script **aldrin** will reside in `/home/paniq/usr/bin`. Run that script.

You will be greeted by a splash screen that fades when Aldrin has been fully initialized. During startup, Aldrin will guess the best sound driver available and run it. If there was a problem during initialization of the sound device, you will be notified of the problem and asked to pick a different configuration in the Preferences dialog.

Pretend that everything has worked correctly. Your startup screen should be looking like this:

Figure 2-1. Aldrin main window after first start.
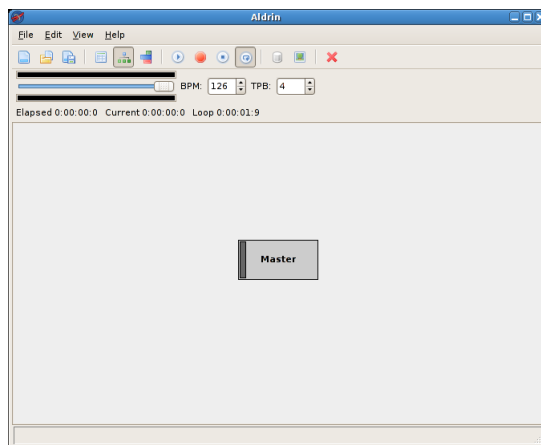
2.3. Configuring Aldrin

Most likely, you will want to configure Aldrin a bit before you get started. If it hasn't opened automatically, open the preferences by selecting View⟶Preferences... from the menu. The preferences dialog will reveal itself to you:

Figure 2-2. Aldrin preferences window (Audio tab).
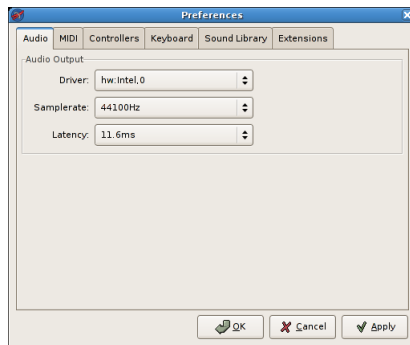
The Audio tab is usually selected by default. It allows you to change settings related to audio output. Following explanation of parameters should you give a better understanding on what to choose.

Device

The output device you wish to use. This usually lists JACK (if enabled), ALSA and OSS output drivers available on your system. All of these drivers require exclusive access to the soundcard for performance reasons, so make sure you have no other sound applications active in the background. I recommend ALSA to be tried first, then JACK, then OSS.

Samplerate

The samplerate at which to render audio in Hz (impulses per second). 44100Hz is usually a good setting (CD Quality). The higher your samplerate is, the more performance Aldrin consumes to render sound. With a lower samplerate, Aldrin consumes less CPU time, but the sound quality is degraded.

Latency

The time between the rendering of a signal and the actual output on the soundcard. Latency controls the size of the internal audio buffer that data is being rendered into before it is sent to the soundcard. A low latency value will give you a better response time, while a higher latency value gives you better performance. A value of 5-10ms should be acceptable.

Once you have set up your favorite settings, hit Apply to test your new settings. If everything works correctly, move over to the Sound Library tab:

Figure 2-3. Aldrin preferences window (Sound Library tab).
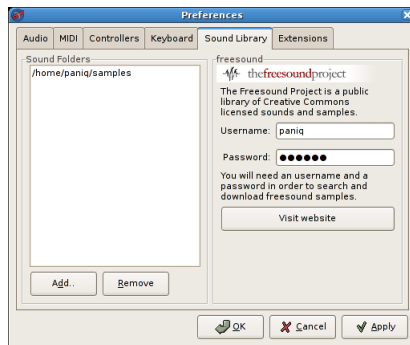
The Sound Library window allows you to manage a list of sample repositories. In order to access samples on your harddrive, you should at least add one sample repository. Click on Add..., then choose a folder to add.

If you want to download samples online from within Aldrin, you need to supply your freesound credentials. If you don't have a freesound account yet, visit the website and get yourself one.

If you are using external MIDI gear, you should have a look at the MIDI and Controllers tabs. They allow you to select MIDI input/output devices and configure MIDI controller bindings.

Aldrin allows you to use your keyboard like a 2-octave piano. Depending which keymap you use, you need to select a proper keymap on the Keyboard tab:

Figure 2-4. Aldrin preferences window (Keyboard tab).
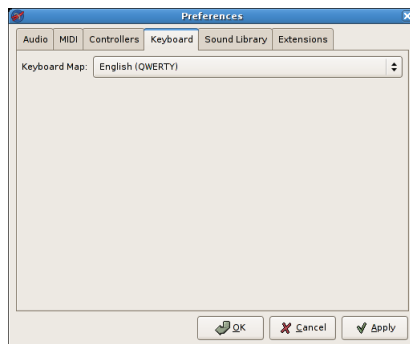
These are the most important settings. Once you are done adjusting, hit OK to apply your preferences and close the preferences window.
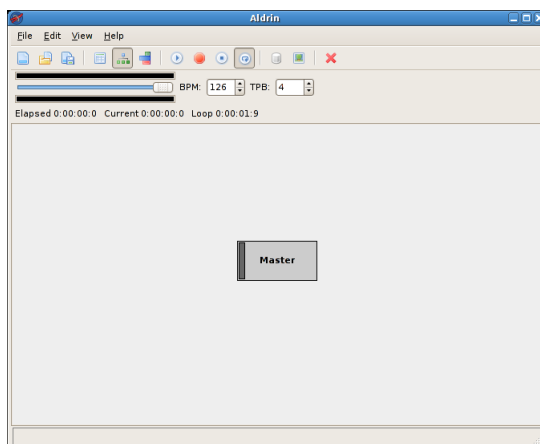
2.4. Making Some Noise

Now that Aldrin is set up, you are ready to create a very small song, consisting of one plugin, one pattern and one sequence. This is a repeating process for every new song, and every new instrument you introduce.

## 2.4.1. Right Where It Belongs: Routing Plugins

You should now be back where you begun:

**Figure 2-5. The router view.**

What you are seeing is the Router View of Aldrin. It allows you to chain generators and effects to define a processing order of signals. What does that mean? Let's try an example.

Right-click somewhere in the empty space of the view. You should see the routers context menu. Select Makk M4wII. A new generator plugin will appear at the coordinates where you right-clicked the view:

Figure 2-6. A new plugin.
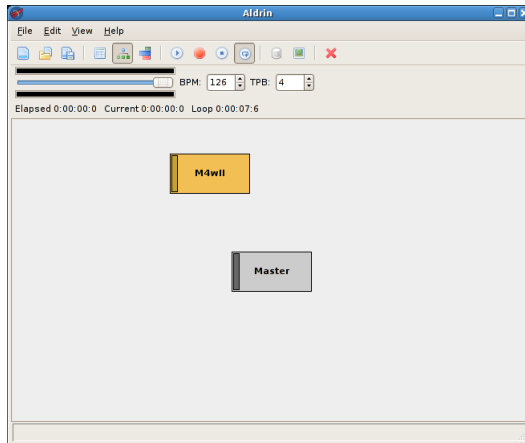
The Makk M4wII is a quite feature-rich synthesizer, imitating an analogue synth model. Right now, the M4wII is not connected to the master output, and thus you will not be able to hear anything. Connect the synth to the master by clicking the plugin with the middle mouse button, and dragging the appearing arrow to the master plugin. You can also connect machines by holding **Shift** and using the left mouse button.

Figure 2-7. A router connection.
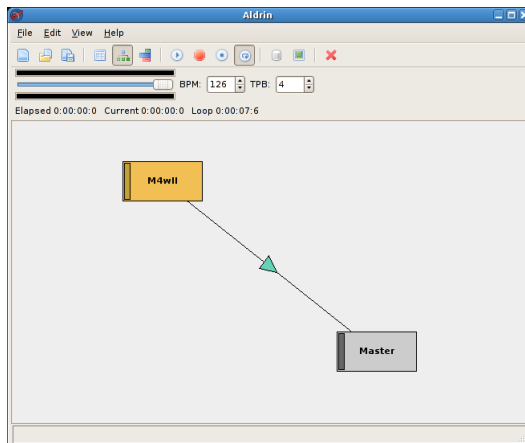
2.4.2. The Core of Composition: Editing Patterns

Now that the synth is created, hit the **F2** key to show the Pattern View.When you create a generator, Aldrin automatically adds a first empty pattern for you to work on, so you should now see the empty pattern:

Figure 2-8. The pattern view.
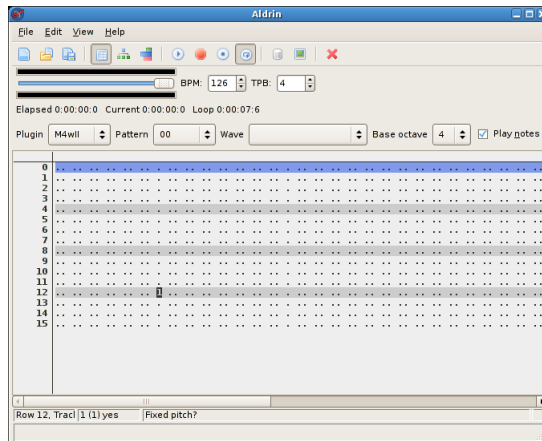
In this pattern, time flows vertically, while actual changes in parameters are arranged horizontally. When a parameter does not change, it's associated column only shows one or more dots. You can click on fields or use the arrows on your keyboard to navigate in the pattern. For each column, the status bar will display the row index, the column index, the current value, it's readable representation, if any, and a long description of the parameter.

It is possible that you can not see the entire pattern width, so you need to move to the first track. Hit **Tab** to jump to the first track, then use the keyboard to enter a few notes into the first column named Note:

Figure 2-9. A pattern track with a few notes entered.
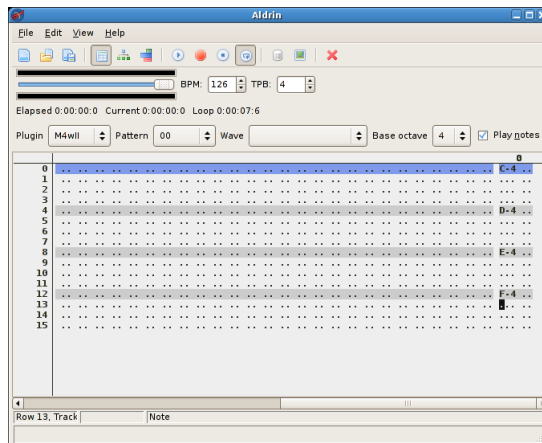
As you enter each note, M4wII will be triggered to play that note, allowing you to preview what you are writing. The cursor travels down by one row. You can change the pattern length and name either by selecting Pattern Properties... from the pattern views context menu, or using the keyboard shortcut **Ctrl**-**Backspace**.

Figure 2-10. Pattern properties.
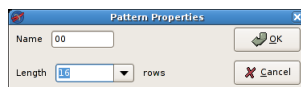
Name

>	The name of the pattern. Aldrin usually enumerates patterns with a sequence like 00, 01, 02, and so on, but you are free to choose your own names.

Length

>	The length of the pattern in rows (= ticks). You can see from the TPB edit field at the top of the main window that the song is currently set up to 4 "ticks per beats", which is the number of rows that is being played per beat. One bar consists of four beats, so that's 4x4 rows = 16 rows - precisely the number of rows that your pattern has right now. The dropdown box provides you with a set of defaults for different speed and bar settings.

2.4.3. Putting It All Together: Sequences

Once your pattern contains some notes, you will want to put your pattern into your songs sequence. In order to do that, hit **F4** to enter the Sequence View. You will see a screen that looks like this:

Figure 2-11. Sequencer View.
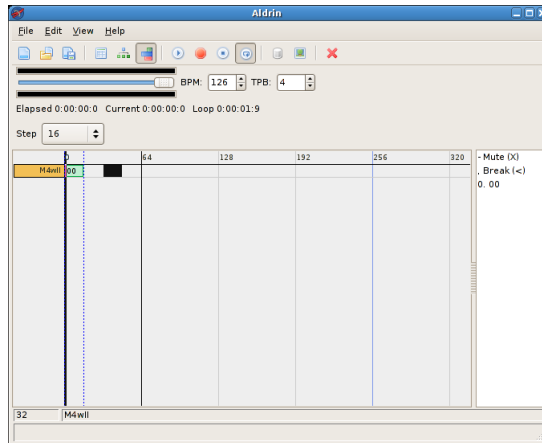
Since you created a generator, the sequence view already contains a track for the M4wII, but it is currently empty. By moving left and right with the **arrow keys** on your keyboard, you can change the time index. Moving up and down will change the track and plugin you are adding patterns for. On the right side of the window, there is a Pattern List, which lists the available patterns you can put into that track. Each pattern is prefixed by the name of the key that you should press on the keyboard to place that pattern into the sequence. In our example, pressing **0** will put pattern 00 in the current track at the current time index your cursor is at.

Now hit **F5** to listen to what your song sounds like. **F8** will stop the song. Your first track - although a bit bland and simple - is finished!

Chapter 3. Keyboard Shortcuts Reference

3.1. Universal Shortcuts

These keyboard shortcuts can be triggered from all locations in the application.