

azMidi

User Manual

Copyright (C) 2004-2006

Exclusive intellectual property of:

Alan Power

All rights reserved.

alan@alanpower.it



This page is intentionally blank.



Introduction

azMidi is the complete MIDI control software for use with all MIDI and audio modules in "live" performance and studio work. It is used with a PC (tablet or laptop and touch screen, tactile screen, external, etc are advised) in "touch" or "keyboard" modes (also with keyboard, keypad, numerical), and replaces any type of MIDI controller or pedal array for the transmission and the reproduction of MIDI messages and AUDIO in real time. azMidi it is an absolute innovation, meaning that no commercial software will perform the same functions as azMidi.

Characteristics

- Controls any type of Windows compatible MIDI card
- Audio control cards too (any Windows compatible sound card)
- Minimal resource requirements of the PC are extremely reduced (PC Celeron with 32 MB of RAM and Windows 98/SE minimum configuration)
- Minimal hard disk requirements (3 MB of program space)
- Stability and advanced reliability for any MIDI pedal-key parameters
- Complete programmability of the main azMidi functions

Principal azMidi features

Complete programming of the main azMidi functions via software that sends MIDI messages (Program Change, Control Change, System Exclusive, On/Off Notes) and audio effects (in WAV format) in real time with the press of a single push-button (from a keyboard or touch screen).

Program sets through simple text files calling up "SONGS", every one of which containing settings, commands of various kinds, steps (configuration changes), macros, links to others songs.

- DEVICE Management (devices you connect to MIDI) global or local
- VARIABLE Management, global or local, allows you to give logical names to devices, MIDI programs (Program Change, Controller), audio clips, etc.
- FAST COMMAND Management, to assign frequently used functions to any push-button for fast, flexible functions and in non-sequential patterns
- STEP Management, each can contain any commands of your choice
- LYRICS or Parts Support, to display for other viewers or to view yourself in real time (including the page changes) in HTML format with support of graphical objects (as you configure, even your own scanned hand written charts, etc.)
- SONG Management, with pre-defined directory of files to execute, with jump possibility to "**automatically sequence**" from one file to another using a predefined ladder (adapted to the entertainer's performances)

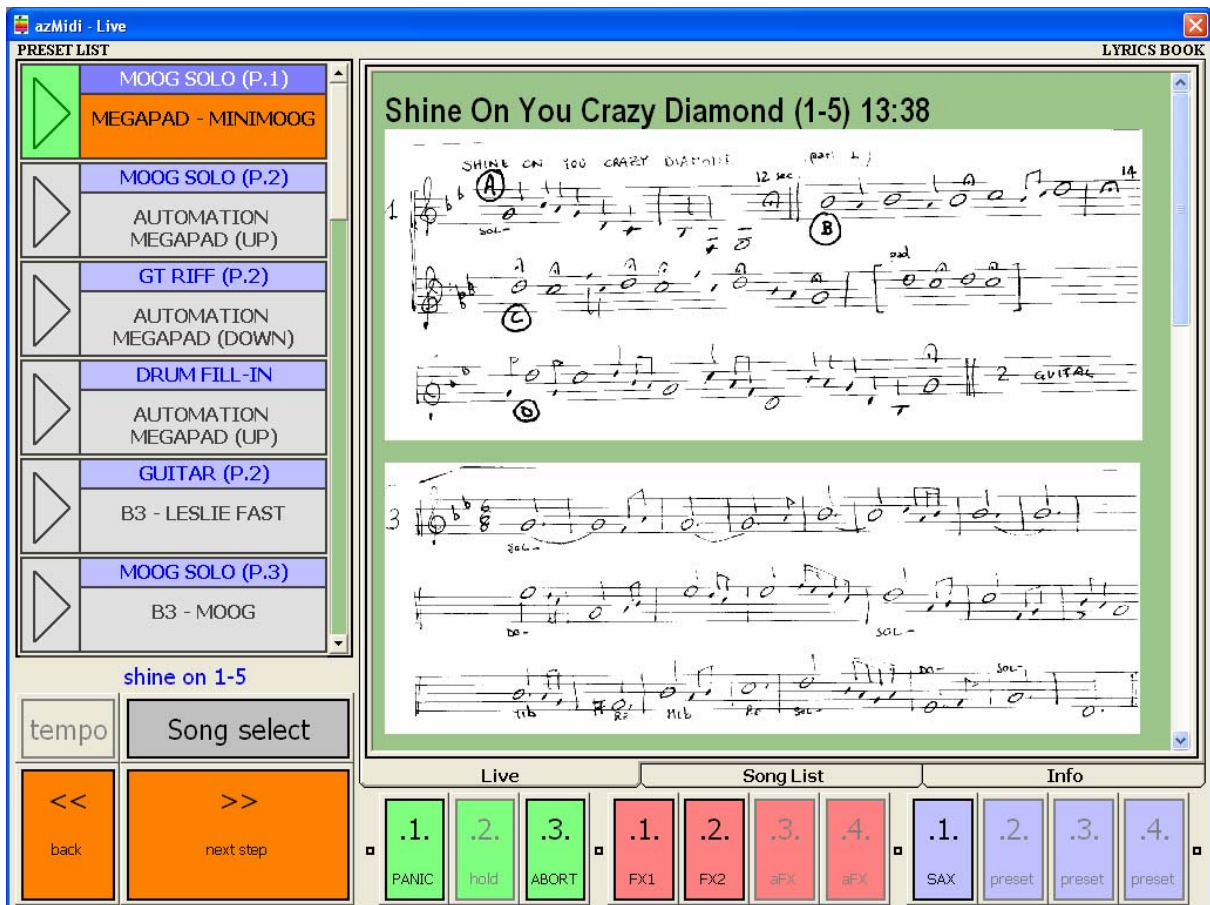


azMidi

User Manual –English

- MIDI AUTOMATION Management automatically sends variable Control Change messages (like progressive volume increase, or the variation of oscillation of a parameter, LFO, etc.)
- DELAY Command Management, sends a certain MIDI command or audio after a time programmable by the pressure of an assigned push-button (assures the execution of a preconfigured "setup change" when you're doing something else)
- Management of standard initialization files, to predefine the DEVICE, the parameters (like the names of the sounds or the macro), in individual files, independent of songs
- MACRO Management, to group multiple commands into individually combinable one STEP commands or a FAST COMMAND
- SETUP Profiles Management, to modify the execution mode according to the profile selected (e.g. live or studio, it determines the instruments, combined instruments, etc.)
- Metronome management is visually programmable, through a simulated LED that flashes according to the TIME predefined for the specific song
- Parsing and control of synthesis before and during the execution of the set is one of the main features of azMidi:

The main screen of azMidi:





SONG Files

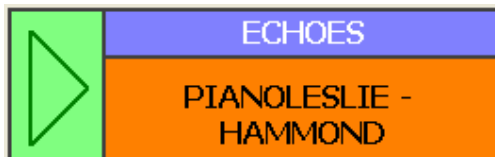
SONG files are standard text format files (with TXT extension) that internally contain the information on the execution structure of the set.

SONG files are user programmed with a structure similar to a PC program. The file "Fat Old Sun.txt" can be viewed to see an example of a song file.

The structure of a song file is as follows:

- The **Head**, (default STEP) contains information to immediately load and messages to send for loading a song. It includes the title and preliminary instrumentation settings, but can also include more information.
- The **Step**, the commands for the setup changes relative to the several portions of the set (e.g. change from piano to violins at the corresponding passage from verse to refrain)
- The Recognition (commands for your SETUP, NEW SONG, EXEC, QUIT) that allow the song to modify the behavior in real time respective to the regular structure
- The **Macro**, program combining multiple commands into one command

When a SONG is loaded, the head (default STEP) is loaded and executed, and it corresponds to the first STEP in the STEP LIST.



The text above is only an example; the instructions relative to the previous STEP are "parsed" (verified) and controlled, but they are not yet executed. Moving to the next STEP (by pressing the "NEXT STEP"

push-button), it executes the first STEP in the SONG. Therefore the instructions contained in it are all executed in the exact sequence they were inserted into the SONG. Pressing "PREV STEP", it's possible to return to the previous STEP. *You* determine the relative execution of the commands.

Imagine having a keyboard with buttons for the programmed sounds in sequence:

- Button 1 = the first sound that is used
- Button 2 = the second sound that is used
- Button 3 = the third,

therefore, in the exact sequence of execution. Imagine now that every time that you activate button 1, its sound is activated.

Thanks to this example, it's easy to understand that azMidi button "1" of the keyboard corresponds to the default STEP (head of the SONG), while the other buttons correspond to the successive STEPs in the exact sequence of your programmed SONG.

Note: If there is no SONG file present in the LYRICS folder, the program cannot be started.



Auto-Configuration Files

Auto-Configuration files are executed upon starting the program (therefore once individually at the beginning), and therefore comfortably load all information immediately that typically does not change during the performance (i.e. names of DEVICE, SOUNDS, MACROS, etc.).

It is naturally possible to leave Auto-Configuration file empty, but that would be inconvenient; since in every SONG file, it would require the insertion of all the previously mentioned necessary default parameters, with consequent performance degradation during SONG loading. The structure of the Auto-Configuration file is identical to the structure of the SONG, with the only difference that the Auto-Configuration files cannot contain an unknown STEP beyond the default.

It is easy to conclude that the Auto-Configuration files are not designed to have "predefined times" for sending commands (STEP); therefore internally, the commands will all be interpreted immediately.

By default, the Auto-Configuration file is called "autoexec.ini".

Profile SETUP

azMidi SETUP profiles preview various SETUP profiles that allow the musician to adapt the behavior of the program according to the configuration that is used. This is a very natural situation if for example devices are used (sound modules) that differ between rehearsals and performances, or if you would just like to "test" a new instrument, without compromising the structure of the performance that is normally used.

Internally, within the Auto-Configuration file and every SONG file, it is therefore possible to *preview* a set of *interpret only commands*; if it is decided to operate with a pre-determined SETUP profile.

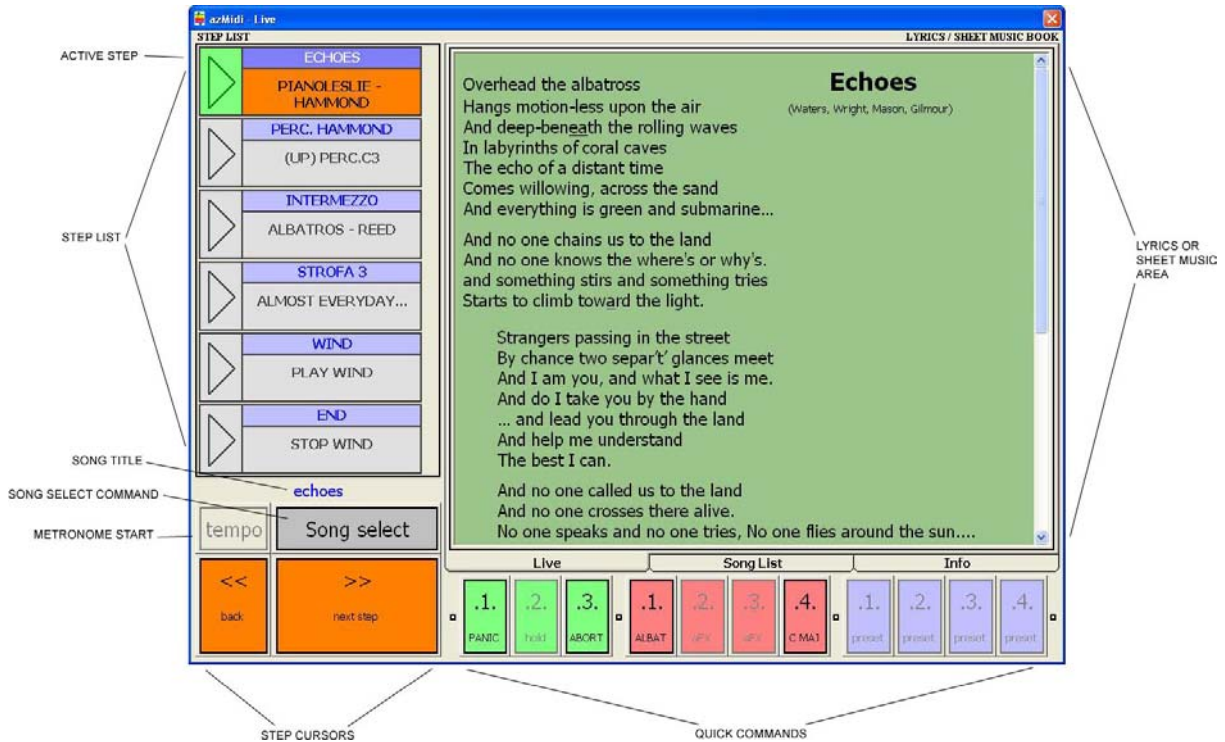
All commands without a specifically associated SETUP profile will always execute unconditionally. For example, we could have a piano sound called SLOWLY that corresponds to a specific program (Program Change) of a specific module (DEVICE), that is valid for any type of session (since the module used is always the same one); while we could have a sound called ORGAN associated to program 10 of the DEVICE D1 with setup "LIVE" and having instead the same sound called with same name (ORGAN) associate to program 68 of the DEVICE D2 with setup "TEST" (since probably for the test, a different module is used to produce that sound).



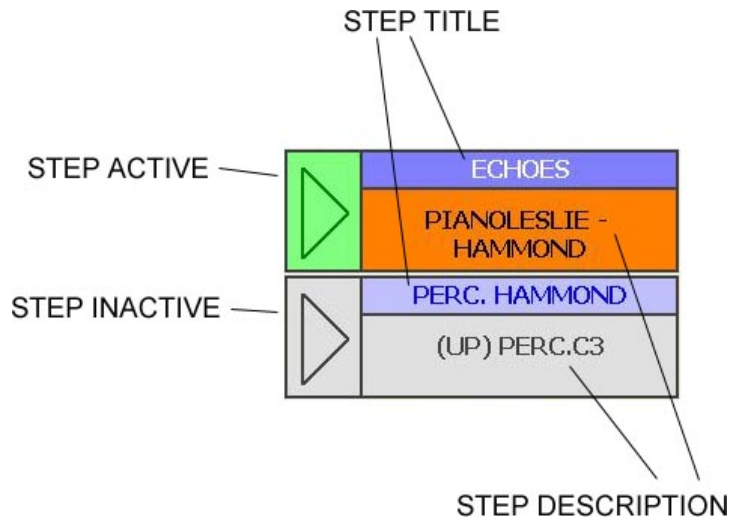
How azMidi works?

azMidi is designed to have a single (full screen) dialogue window, where all necessary controls for the performance and operational test are present. Auto-Configuration files and SONG files must be prepared with an external editor (e.g. Notepad or another similar editor).

We'll take a glance at the main window:



You will notice that it is designed with several STEPS:





There are 3 types of FAST COMMANDS

The **HOLD** command, the **AUDIO** command and the **PRESET** command.

In addition to the names of the commands, it is important to understand that the command associated to every button is fully customizable (like also the name of the same button, as seen in the figure above). That means that it is possible to associate a MIDI or LYRICS command to an AUDIO button.

The **HOLD and PRESET** commands act in the same way:

Pressing a button makes it possible to associate a command, and release another command (adapted for example for short temporary effects mostly). These buttons react like a momentary contact switch or a piano sustain pedal. They perform the function only as long as they are pressed.

The **AUDIO** type commands instead, once pressed, remain "pressed" or "latched" (blue color), and in order to release, it is necessary to press them again. For this reason, AUDIO push-buttons are adapted to associated audio commands like "PLAY" (activation = first press) and "STOP" (deactivation = second press). It is however possible to associate commands of any type, also MACROs (commands that contain multiple commands).