

README.WRI



>>> MidiThru version 1.0 -- copyright © 1993 by David Pannett <<<

Overview:

This nifty little MIDI utility lets you patch MIDI input directly to MIDI output under Windows 3.1. I needed such a utility; you probably do too. You can select from among all installed MIDI devices on your system, e.g. patching the input from the MIDI IN port of an MPU-401-compatible MIDI interface to the output driver for an internal synth. The MIDI Mapper is also supported as an available output device.

MidiThru is useful in many situations:

- To control your (internal or external) MIDI output devices from any MIDI controller via any MIDI input port supported by a Windows 3.1 driver. If your system is configured so that not all of your (internal or external) MIDI output devices can be controlled by your (perhaps multiple) MIDI input devices except via your computer, this program provides a simple and direct solution.
- To switch among available MIDI devices without physical repatching.
- To play on one (internal or external) MIDI output device while another program such as a sequencer plays on a different MIDI output device.

If you have an internal synth on a sound card, a dumb (silent) MIDI controller device, and/or multiple MIDI output devices at the system level (for example, a sound module accessed via a MIDI output port as well as an internal synth on a sound card), you need a utility like this to help you get maximum use out of your system. Just use MidiThru to connect any input device to any output device and you're in business, whatever your setup.

*****PLEASE READ*****PLEASE READ*****PLEASE READ*****PLEASE READ*****

Installing MidiThru:

MidiThru requires the file BWCC.DLL (dated 6/10/92 or later). It's not unlikely that you have this file already installed in your WINDOWS\SYSTEM directory; if not, the file is widely available online. BWCC.DLL, which is used by many shrink-wrapped as well as shareware programs,

contains functions supporting the Borland Windows Custom Controls.

Otherwise simply create a directory for MidiThru and copy the files MIDITHRU.EXE and MIDIPUMP.DLL into it.

To add a MidiThru icon to a Program Manager group, follow these instructions:

1. In Program Manager, select the group in which you want the icon to appear.
2. Select "New..." from Program Manager's File menu.
3. In the dialog that appears, click on "Program Item", then "OK".
4. In the "Program Item Properties" dialog that now appears, fill in the "Description" field (the label for the icon, e.g. "MidiThru") and the "Command Line" field (the full path to the executable file, e.g. "C\MIDITHRU\MIDITHRU.EXE"), then click on "OK".

An icon for MidiThru will appear in the selected Program Manager group.

*****IMPORTANT*****|IMPORTANT*****|IMPORTANT*****|IMPORTANT*****

Notes on MIDI device conflicts under Windows 3.1:

In general, two MIDI programs running under Windows cannot use the same MIDI device simultaneously. There are some exceptions to this rule, however; for example, one program might use the MIDI input port on a physical device at the same time as another program uses the MIDI output port of that device. To some extent this depends on the software drivers used to control the physical device.

However, the degree of cooperation exhibited by your other software is also a factor determining what sharing of MIDI devices is possible. Some MIDI programs running under Windows use their own drivers and simply hog the physical MIDI device whether or not they are using it at the time. While such an antisocial MIDI program is running, no other program will be able to access the same physical MIDI device. In some cases, other MIDI programs cannot access the same physical MIDI device used by an uncooperative MIDI program even after the uncooperative program has been removed from memory (Finale 2.0.1 is an example.) Using such MIDI programs together with MidiThru will normally succeed only if each uses a different MIDI device, although a limited form of device-sharing may still be possible in some cases; see below.

Some MIDI programs that do use regular Windows drivers are equally guilty of hogging the selected MIDI devices as long as they are running. Again, your mileage may vary in attempting to use MidiThru concurrently with such programs. If each uses a different device no conflict should arise, and it will often be possible to output from a second program to the same physical device simultaneously used for input by MidiThru (so long as MidiThru sends its output to a *different* physical device), even if the second program is not particularly cooperative. Adjusting the settings of the second program may be necessary to achieve even this limited cooperation.

For some devices, there exist special MIDI drivers permitting concurrent device access by multiple clients. If you are using such drivers, device conflicts may not be an issue. Otherwise, attempting to access an input or output device that is currently claimed by another program will result in an error message from MidiThru. Under which conditions another program may lay claim to a device or device driver depends on that program, and perhaps on its current configuration. MidiThru itself is written to be as cooperative as possible.

*****REGISTER*****REGISTER*****REGISTER*****REGISTER*****REGISTER*****

Registration:

If you find this program useful, you should register it. In return you will receive an enhanced version of this program on diskette. The registered version includes, among other features, a real-time 16-channel MIDI mixer utility.

To register, send \$5.00 US funds to:

David Pannett
9 Lovewell Road
Wellesley, MA 02181 USA

Please indicate whether you prefer 3.5" or 5.25" floppy disks.

Please direct comments, questions and suggestions to any of the following electronic addresses:

Internet:
dpannett@lucy.wellesley.edu

GEne:
D.PANNETT

America Online:
DavidP5429

Many thanks for any and all comments.

David Pannett
11/7/93