ThisDrivel

… I just churned out THIS DRIVEL:

« Configuring MIDIsport 8×8/s MIDI ports using a Windows 2000 VM

More on configuring MIDIsport 8×8/s MIDI ports

Further to Configuring MIDIsport 8×8/s MIDI ports using a Windows 2000 VM, I've found the software/hardware communication to be very flaky, but I'd read that this was always the case.

After some failures, I found that trying to change port mappings and immediately use the hardware didn't work for me while connected to the PC, using only external MIDI devices at least — the mappings didn't seem to change and the unit wasn't showing any activity.

Important: Note that **I'm mainly using the MIDIsport 8×8/s as a standalone device** with USB mode permanently selected on the front panel, and using a VM per the above post to configure its MIDI port mapping. I haven't yet tried using it in conjunction with a DAW since getting port mapping to work.

Key point of this post, for standalone usage:

Shut down the VM after configuring and sending mappings to the MIDIsport, and disconnect the USB cable from the MIDIsport. It should then show LED activity with the current mapping after a few seconds (5-10 in my case).

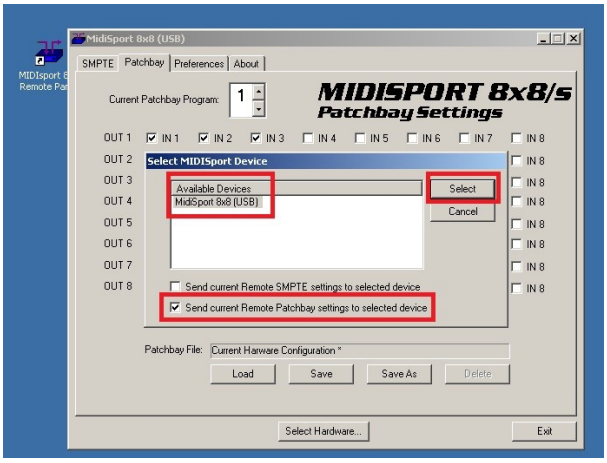
Workflow

So far, my workflow to change ports appears to be this:

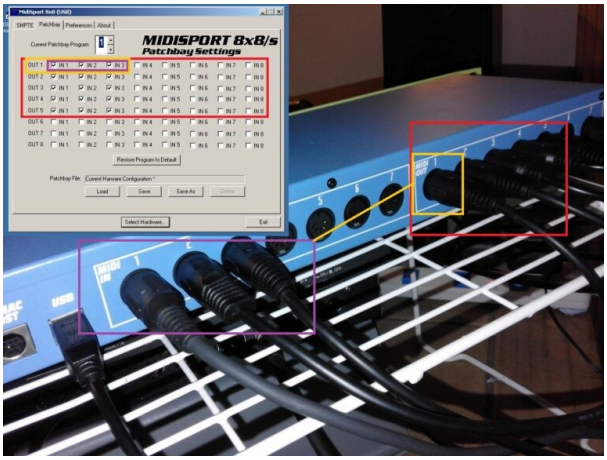
1) With MIDIsport 8×8/s running and connected to the PC, run the virtual machine;
2) Once up and running, take over the MIDIsport USB connection through the VM as per the earlier post;
3) Configure the ports as you wish via the Remote software;
4) **Send the port configuration to the device** (see below);
5) Exit the Remote software and SHUT DOWN THE VM;
6) For standalone use, disconnect the MIDIsport's USB cable from the PC and wait 5-10 seconds for its LEDs to update*.

** Note that I haven't yet tried to use with a DAW at this point, so not sure how that would work.*

Regarding item 4), I don't believe the port mappings are saved until you **manually** send them to the device, like so:



As far as setting up the ports go, it's a bit of a mind-bender, but **you are basically saying that for each of the physical MIDI Out ports, you want to receive data from the selected MIDI In ports only**:



(Click for full-size image.)

In the above example, viewing the hardware itself, I have 3 physical MIDI inputs (highlighted in purple) and 5 MIDI outputs (highlighted in red) — please ignore the cable plugged into MIDI OUT 7 here!

For OUT 1 in the software, highlighted in amber in both hardware and software photos, I have selected the 3 MIDI Ins — so MIDI Out port 1 will receive input from my 3 controller devices plugged into MIDI IN 1, 2 and 3.

Compare the hardware and software images to understand how all 3 MIDI Ins are routed to physical MIDI Out port 1.

Once you 'get' this, the same has been repeated here for all 5 outputs — so each output MIDI device will receive input from all 3 input MIDI devices.

You can of course adjust to suit your setup, eg. you may only want MIDI Output 5 (let's say this is a simple synth) to receive input from your master keyboard, connected to MIDI Input 1; in this case you would select only IN 1 against the OUT 5 line.

It's important to note (this is where it really starts to blow one's mind) that the port numbers are not MIDI channel numbers! This all affects the physical connections only — the device plugged into MIDI In 1, for example, may have MIDI channel 5 selected.

As long as the physical cables between the input device on channel 5 and the output device on channel 5 are mapped to 'see' each other, channel selection just works as normal.

Note that the MIDIsport 8×8/s can save 8 separate mappings, under "Current Patchbay Program" — this

is program 1.

Note also that you can store your current mappings by clicking Save and inputting a name for the mapping. I believe to load you would hit Load, select a saved mapping, then Select Hardware to send the mapping to the device, per ‘Workflow’ item 4, above).

You select the programs on the hardware by holding the MIDI Reset button for a second; this shows the current selection by lighting all LEDs *except* the current program. (The current program will be unlit.)

Holding for longer will cycle through the programs, 1 to 8; keep it held until your desired program is unlit, then release.

Finally, as noted, once you’ve changed your mappings and sent to the device (per ‘Workflow’ item 4, above), **shut down the VM** to give USB control back to the main PC, and disconnect the USB cable from the MIDIsport. (Again, I have only explored standalone operation, so not yet sure how/if it works with a DAW… though it should.)

I did get into an odd state at one point, whereby attempting to reconnect the MIDIsport to the VM kept dropping the USB connection — my only solution was to reboot the host PC and MIDIsport. Make sure the MIDIsport is up and running *before* re-running the VM and connecting to the MIDIsport. Also note that the MIDIsport is notorious for needing to be running before external MIDI devices connect to it, so you may need to restart everything!

However, once you have set up a mapping that works, in most cases you probably won’t need to use the VM — just jump through the above hoops if you need to change the mapping.

After that, it should just work standalone with USB mode selected on the front panel and the USB cable *disconnected*. Make sure you’ve selected the right program number on the MIDIsport, as per the process above. It should be the one you edited in the software!

In short:

1) MIDIsport running first;
2) Run VM;
3) Configure ports;
4) SEND config to MIDIsport;
5) SHUT DOWN VM;
6) Disconnect MIDIsport’s USB cable;
7) MIDIsport should then work as a standalone device!

Items 4 and 5 are important after each remapping — it probably won’t work until you do this.

Again, my use case is currently running external devices/DAW-less only!

This entry was posted on Saturday, January 9th, 2021 at 02:58 and is filed under Uncategorized. You can follow any responses to this entry through the RSS 2.0 feed. Both comments and pings are currently closed.

Comments are closed.

ThisDrivel is proudly powered by WordPress
Entries (RSS) and Comments (RSS).