

## **ABOUT STANTON'S SCS.1 DAROUTER SOFTWARE**

The Stanton SCS.1 DaRouter software serves a couple different purposes. First, it's the "engine" that allows communication between the SCS.1 hardware and your performance software. DaRouter also allows users to reconfigure the layout of their SCS.1 controller with the use of "presets". Presets typically end with the .btme extension, and can be found in the Stanton SCS.1 DaRouter folder on your computer.

This guide will walk you through the features of the SCS.1 DaRouter software, and explain how to use them.

## The DaRouter Interface

Let's start at the top of the program and take a look at the **Active Preset** drop down box.



This is where you select the preset you want to load. All of the officially released presets should be located here. If you want to load a preset that doesn't appear in the Active Preset drop down box, you can either drag and drop the preset file onto the DaRouter interface, or you can select BROWSE from the drop down box.

Next is the **Restart** button.



The Restart button restarts the preset in the ACTIVE PRESET box. This can be helpful if you're experiencing issues with the preset or hardware. Remember that this completely restarts the preset, and pushes initial values to things like the pitch slider – so use with caution

Finally, we have the “?” button.



This button will appear solid when a help file is available for the selected preset. If there is not a help file, it will be transparent. Pressing the button will open a locally stored web page that explains all the functionality in the preset. It's like having a manual for the preset available whenever you want.

Now let's look at the next section of DaRouter – the **Device Configuration** section.



This area tells you what hardware is recognized by the SCS.1 DaRouter software, and gives you some options for its use.

The first thing to check is that hardware is correctly recognized. In the example below, you can see that even though a preset is loaded, no hardware is detected.



See the drop down box under each SCS.1 device? If it says “detecting device...”, that means that the SCS.1 DaRouter software doesn’t see any hardware connected. Check to see that your Firewire cables and power adapters are connected properly, and that your hardware is turned on. If the hardware looks like it’s connected correctly, and you’re running a Windows computer, make sure the ASIO driver is correctly installed as well. In OSX, this is not necessary because the SCS.1 devices are CoreAudio compliant, but check your audio and MIDI devices to make sure your SCS.1 controllers are recognized there.

When your SCS.1 hardware is recognized by DaRouter, it will look like this...



You can see that in the above pic, there are serial numbers under each detected device (I only have one of each controller connected). There will also be a glow around the controllers that are attached. This is what you want to look for when you run DaRouter.

The final area in the SCS.1 DaRouter software is the **Activity Monitor**.

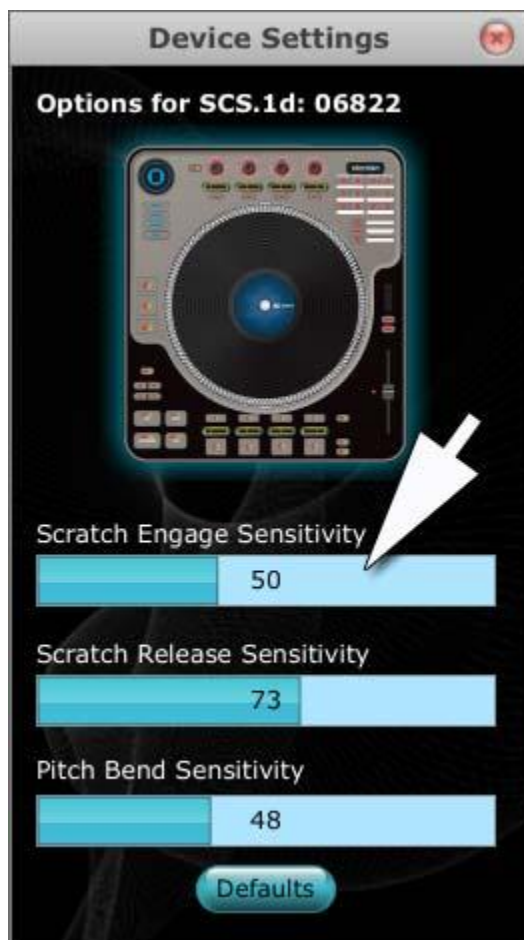


These 3 symbols actually let you know what's going on in DaRouter. The first arrow shows MIDI traffic going INTO DaRouter. The square in the middle shows when preset rules are being processed and DaRouter is doing its job, and the last arrow indicates MIDI traffic OUT OF DaRouter.

Now that we've covered the main features in the SCS.1 DaRouter software, let's look at the options available for each of the devices, starting with the SCS.1d platter controller. You get to the options panel by pressing the OPTIONS button.



This will open the OPTIONS panel for the SCS.1d. The first parameter we have access to is the **Scratch Engage Sensitivity**.



This parameter tells the SCS.1d when to transition to a scratch message when the platter is being used. If it's set too low, it will constantly send scratch messages and pitch bending will be impossible. Too high a setting, and the platter will transition from a pitch bend to a scratch very slowly. A good middle ground is 50 (which is default). Remember that lower is more sensitive, and higher is less sensitive.

Next up is the **Scratch Release Sensitivity**.



The Scratch Release Sensitivity variable tells the platter how long to “wait” until it turns off the scratch message and resume normal playback. A lower setting means a faster transition between the 2 modes, but having it too low could cause the platter to drop out of scratch mode before it should, making a scratch sound sloppy. The default setting for this variable is 73.

Finally, we have the **Pitch Bend Sensitivity**.



The Pitch Bend Sensitivity variable tells the platter how easily it will change from normal operation to pitch bending. A lower setting means that the slightest touch will create a pitch bend. Higher settings make pitch bending harder to occur. A setting too low will cause the pitch bend message to happen even when the platter isn't being touched (and is obviously not desirable). The default setting for this variable is 48.

If you want to get back to default values, you can simply press the DEFAULTS button at the bottom of the SCS.1d OPTIONS panel.

Now let's take a look at the SCS.1m options. Get there by pressing the OPTIONS button below the SCS.1m.



This opens the SCS.1m audio options. This is your ASIO control panel for the SCS.1m, which is why this option will only appear in Windows.

Our first available setting is the **Sample Rate**.



Sample rate for the audio interface can be changed here. The sample rate is the number of “pictures” of the audio that are taken in a second. Typically, higher is better, but you have to weigh the potential benefit of a more accurate audio reproduction against things like desired file size and available system bandwidth. Sometimes a good middle ground is better than selecting the highest available.

The next variable up is the **Firewire Latency**.



Lower is better, but if you go too low you will start to get clicks and pops in your audio playback. If you set this too high, the playback will lag behind and you'll perceive a delay between an event (like pushing a button) and its effect on the audio.

Finally, we have the **ASIO Buffering** variable.



Like the Firewire latency variable, ASIO buffering can cause delay between control input and response if set too high. If set too low, it can cause audio playback to pop and click.

The way to determine the best settings for both the Firewire latency and ASIO buffering is to set them at their lowest setting and play a set. Use the equipment and software to the fullest. If you experience pops or clicks in the audio, try raising the ASIO latency a single step, then play a bit more. If there are still some glitches in the audio, raise the Firewire latency a single step. Repeat these steps until you get stable and click-free audio playback.

If you have trouble even at high latencies getting playback to be stable, you will need to do some system tweaking in order to optimize playback.

There are some tips for optimizing playback located in the appendix of your SCS.1 manual. You can also find valuable tips on optimizing your Windows computer for audio playback at <http://www.blackviper.com/>.

## HELP

We hope that this help document gives you all the information you need to get up and running. If you have any additional questions, please feel free contact our Customer Support Monday through Friday at +1 954.316.1500 from 9AM to 6PM EST.

We also have a lot of great information on our forums...

<http://forums.stantondj.com/>

And be sure to check our Knowledge Base for answers to common issues and questions.

<http://www.stantondj.com/activekb/>

## LEGAL

Windows is a registered trademark of Microsoft Corporation in the United States and other countries

OSX is a registered trademark of Apple Inc.