# MULLIGAN

## Hands-On Plugin Control for Pro Tools

# Introduction

## Welcome to a New Plugin Workflow

Thank you for trying Mulligan, the standalone app that enables you to use real knobs to control any plugin in **Pro Tools™**, while using the MIDI controllers of your choice.

At its core, Mulligan connects a bank of **eight hardware knobs** to the currently selected plugin. It also provides a **Heads-Up Display** onscreen, as a transparent overlay window, to show at a glance which plugin parameters are mapped to your knobs.

In addition, Mulligan enhances your workflow with a slate of controls for the **current track** (level, solo, mute, pans, sends), as well as commands for plugin navigation, track navigation, and **transport** control. (You can view the full list of Mappable Commands here.)

The term "mulligan" means a "do-over" or a second chance, and that's how this project started for us – as a second chance for hands-on plugin control, after the Novation™ Nocturn (which we were big fans of) dropped Pro Tools compatibility. Once tactile control becomes second nature, it's hard to go back to just clicking with a mouse!

Our intent with Mulligan was to design a compact, affordable, task-focused tool to enhance your mixing process. We look forward to your feedback helping us to shape future updates. Please keep in touch and let us know about your experience with Mulligan.
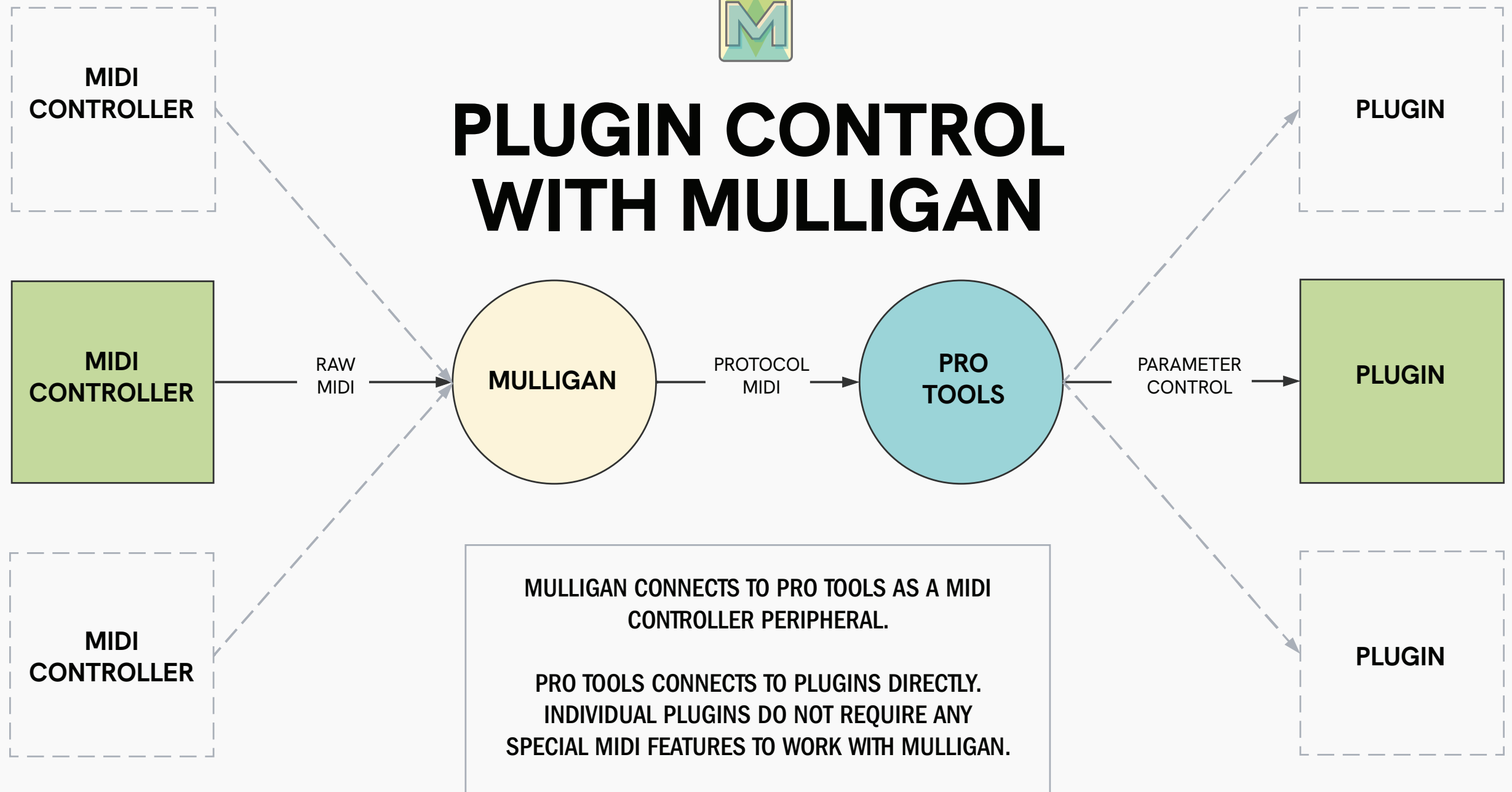
PS: This manual provides an in-depth look at Mulligan's features, but we also have a **Quick Start Tutorial video** to get you up and running in minutes:

https://www.refusesoftware.com/documentation/videos/17

# PLUGIN CONTROL WITH MULLIGAN

MIDI CONTROLLER

MIDI CONTROLLER — RAW MIDI → MULLIGAN — PROTOCOL MIDI → PRO TOOLS — PARAMETER CONTROL → PLUGIN

MIDI CONTROLLER

PLUGIN

PLUGIN

MULLIGAN CONNECTS TO PRO TOOLS AS A MIDI CONTROLLER PERIPHERAL.

PRO TOOLS CONNECTS TO PLUGINS DIRECTLY. INDIVIDUAL PLUGINS DO NOT REQUIRE ANY SPECIAL MIDI FEATURES TO WORK WITH MULLIGAN.

Section 01

# SETUP

# Connecting Mulligan and Pro Tools

On macOS, Mulligan creates its own virtual MIDI ports for communication with Pro Tools.

On Windows, before continuing with these setup steps, you will need to use a utility app (like the free loopMIDI) to create two MIDI ports, and name them **Mulligan In** and **Mulligan Out**.

## Enable Mulligan as a MIDI Input Device

1. Launch Mulligan.

2. Launch Pro Tools.

3. In Pro Tools, choose the menu item:
   **Setup > MIDI > MIDI Input Devices...**

4. In the **MIDI Input Enable** dialog box, check the box next to **Mulligan** (or **Mulligan Out** on Windows).

5. Click OK.

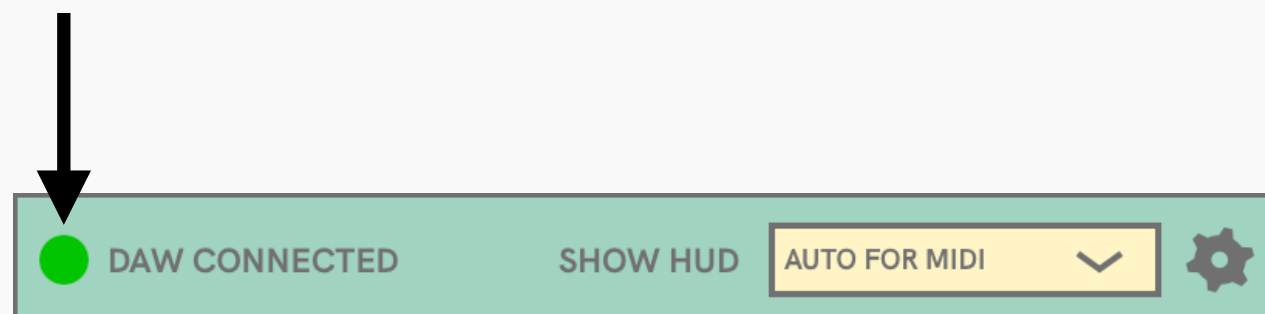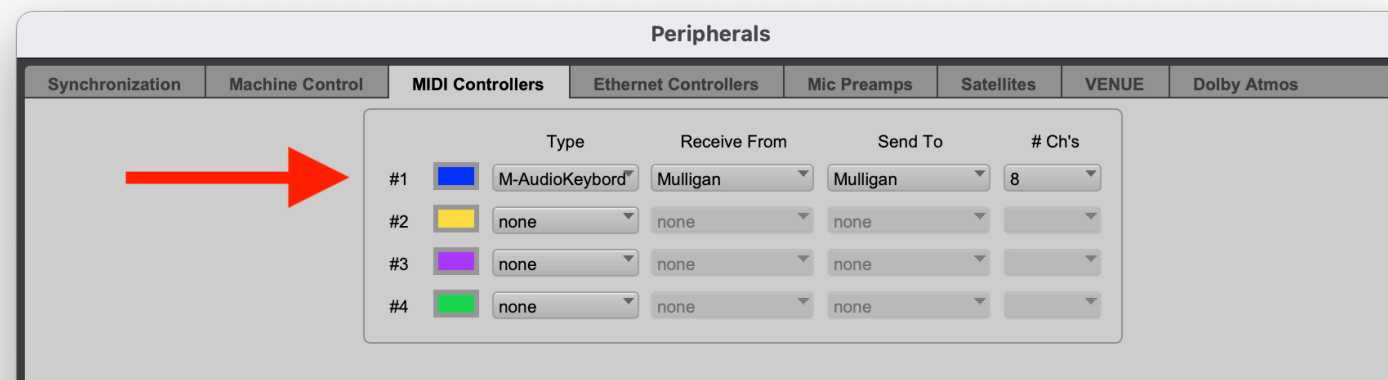**!** Continue to the next page to complete the connection steps.
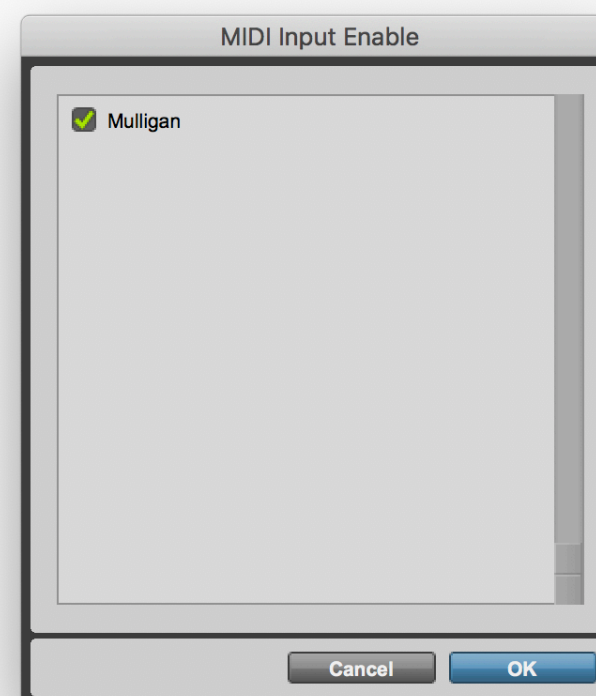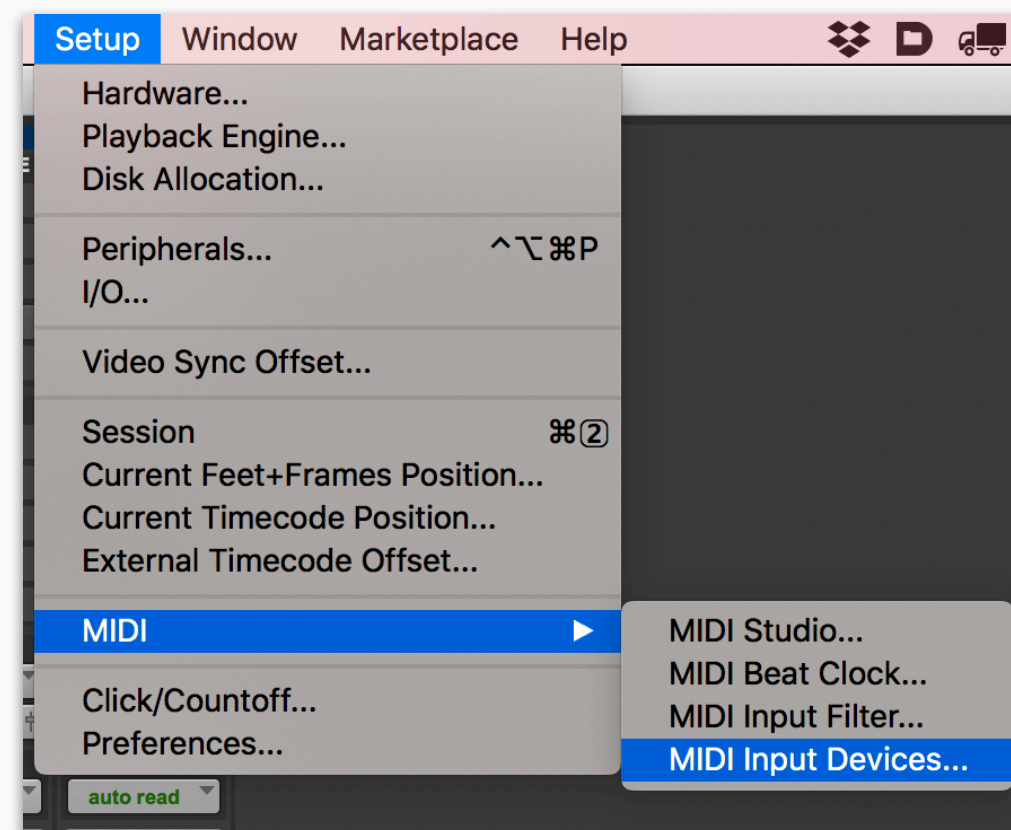
# 01   Setup

## Select Mulligan as a MIDI Controller

1. In Pro Tools, choose the menu item: **Setup > Peripherals...**

2. Click on the **MIDI Controllers** tab.

3. In the first row, select **Type** **M-Audio Keyboard** from the menu.

4. In that same row, for both the **Receive From** and **Send To** menus, select **Predefined > Mulligan**.*

5. Click **OK**.

The setup for connection between Mulligan and Pro Tools is now complete. The **DAW Connected** light in Mulligan should now be green, to show that the connection is active.

\* On Windows, for **Receive From** select **Mulligan Out**, and for **Send To** select **Mulligan In**.

**Peripherals**

| Synchronization | Machine Control | MIDI Controllers | Ethernet Controllers | Mic Preamps | Satellites | VENUE | Dolby Atmos |
|---|---|---|---|---|---|---|---|

| | | Type | Receive From | Send To | # Ch's |
|---|---|---|---|---|---|
| #1 | | M-AudioKeybord | Mulligan | Mulligan | 8 |
| #2 | | none | none | none | |
| #3 | | none | none | none | |
| #4 | | none | none | none | |

**DAW CONNECTED**   SHOW HUD   AUTO FOR MIDI

**!** These are one-time setup steps that you will not have to repeat in the future, as long as you launch Mulligan **before** launching Pro Tools.

If you *do* happen to launch Mulligan **after** Pro Tools, you can just repeat steps 1-3 of *Select Mulligan as a MIDI Controller* to reconnect them. When you click **OK** to close the Peripherals window, Pro Tools will reconnect with Mulligan.

## Authorizing Mulligan

Until authorized with a license file, Mulligan will operate in demo mode. While in demo mode:

- The controller mappings you assign in the Mapping Window won't be saved when you quit the application

- Custom plugin and parameter names you enter into the HUD won't be saved when you quit the application

- Mulligan will timeout and stop sending control messages to the DAW 20 minutes after launching the application

After the 20 minute demo timeout, you are free to quit and relaunch Mulligan. However, you will need to remap your MIDI controller and reconnect Mulligan to the DAW at each launch.

To authorize Mulligan and remove these demo limitations, purchase a license at our website, and then log in to your user account and download a license file.

Once you have your license file, launch Mulligan and look for the gear icon at the top right of the main window, which will open the **Settings Panel**.

Click the **Load License File** button in the Settings Panel, and find the license file that you previously downloaded. Mulligan will now show your registered username in the Settings Panel, and the demo limitations will now have been removed.
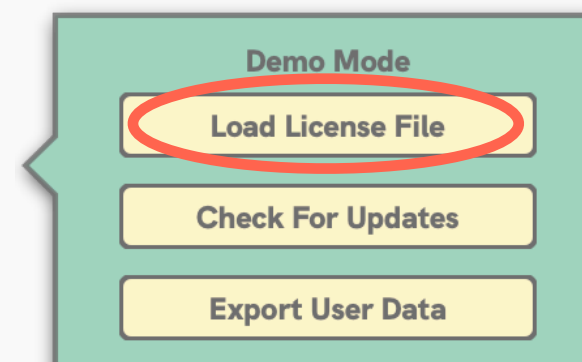
**Section 02**
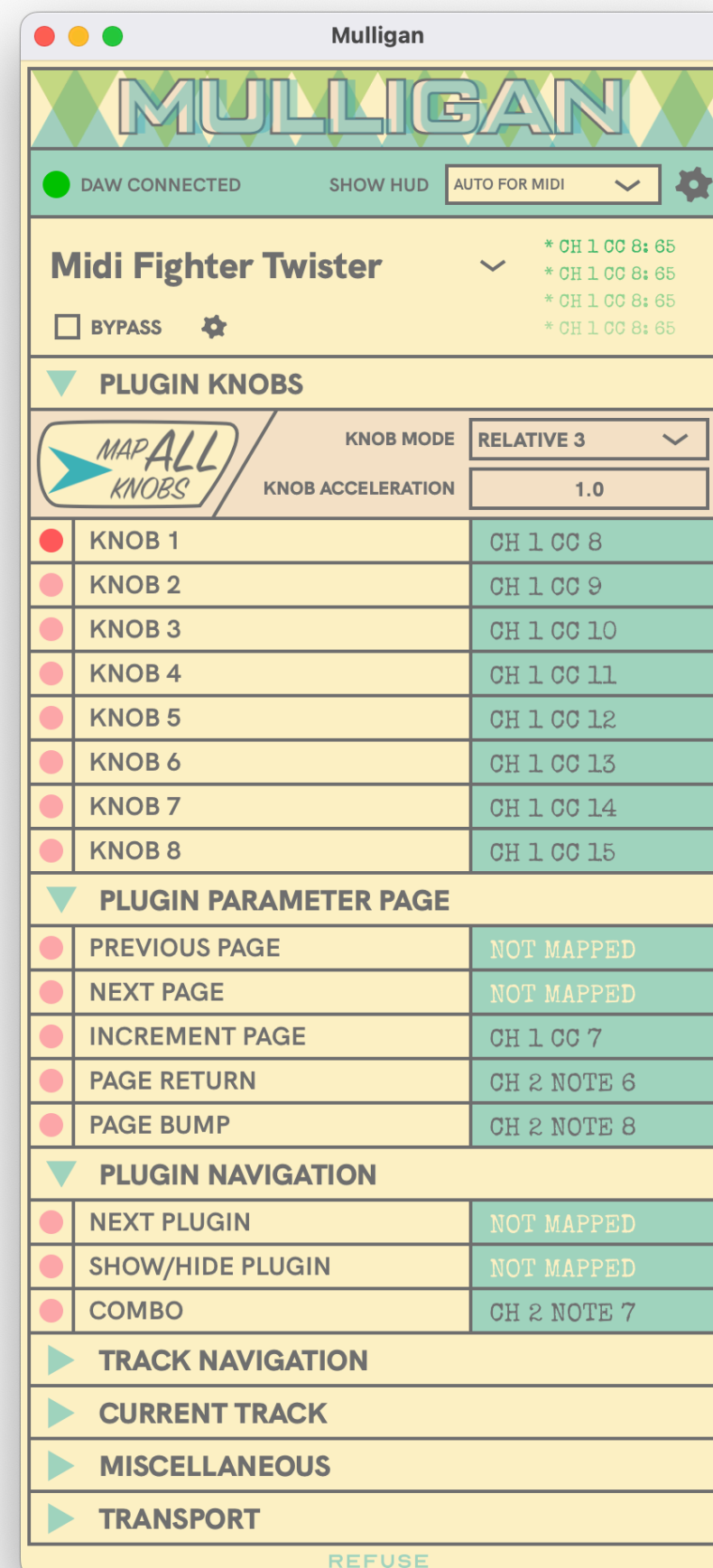
# USE

## The Mapping Window

The **Mapping Window** is where you map MIDI controller knobs and buttons to Mulligan's commands. You can use multiple MIDI controllers simultaneously, and each controller has its own separate map.

Use the **MIDI device menu** at the top of the mapping window to select the current MIDI device to edit.

For your first mapping, we suggest that you start with the **Map All Knobs** button. This allows you to quickly map all eight of the Plugin Knobs. Click the Map All Knobs button, then turn the first hardware knob **slowly to the left** (counterclockwise). Mulligan maps the knob you turned to Plugin Knob 1, and then automatically maps the rest of the plugin knobs to consecutive CC numbers. So with one click and one turn of a knob, all eight knobs in the Plugin Knobs section are now ready to use.

The other commands are mapped individually. The steps for mapping are simple:

1. **Click** a command to select it. The **Mapping Assistant** window will pop up to give additional information about the selected command.

2. **Turn** a knob (**SLOWLY** to the left) or **push** a button on your MIDI hardware. The Mapping Assistant's status

**Mulligan**

**MULLIGAN**

● DAW CONNECTED    SHOW HUD  [ AUTO FOR MIDI  ⌄ ]  ⚙

**Midi Fighter Twister**      ⌄   * CH 1 CC 8: 65
                                 * CH 1 CC 8: 65
☐ BYPASS  ⚙                      * CH 1 CC 8: 65
                                 * CH 1 CC 8: 65

▼  **PLUGIN KNOBS**

| MAP ALL KNOBS | KNOB MODE | RELATIVE 3 ⌄ |
| | KNOB ACCELERATION | 1.0 |

| ● | KNOB 1 | CH 1 CC 8 |
| ● | KNOB 2 | CH 1 CC 9 |
| ● | KNOB 3 | CH 1 CC 10 |
| ● | KNOB 4 | CH 1 CC 11 |
| ● | KNOB 5 | CH 1 CC 12 |
| ● | KNOB 6 | CH 1 CC 13 |
| ● | KNOB 7 | CH 1 CC 14 |
| ● | KNOB 8 | CH 1 CC 15 |

▼  **PLUGIN PARAMETER PAGE**

| ● | PREVIOUS PAGE | NOT MAPPED |
| ● | NEXT PAGE | NOT MAPPED |
| ● | INCREMENT PAGE | CH 1 CC 7 |
| ● | PAGE RETURN | CH 2 NOTE 6 |
| ● | PAGE BUMP | CH 2 NOTE 8 |

▼  **PLUGIN NAVIGATION**

| ● | NEXT PLUGIN | NOT MAPPED |
| ● | SHOW/HIDE PLUGIN | NOT MAPPED |
| ● | COMBO | CH 2 NOTE 7 |

▶  **TRACK NAVIGATION**

▶  **CURRENT TRACK**

▶  **MISCELLANEOUS**

▶  **TRANSPORT**

REFUSE

bar changes from **Ready To Map** to **Mapped**, and the knob or button is now mapped to that command.

If you want to remap a command to a different knob or button, select it and press the **Delete** key to erase the existing mapping. This will then re-arm the command for mapping.

We'll mention briefly here that Mulligan works best when used with encoder knobs sending incremental MIDI data, or what we call "**Relative**" knobs. To learn more about setting up your mappings to get the most out of your Mulligan experience, please see the <u>Optimizing Your Mappings</u> section.

## Finishing Your Controller Mapping

It is important to note that you are not required to map *all* the available commands to your MIDI controller! Mulligan allows you to pick and choose, to customize your workflow in whatever way feels most intuitive to you.

In addition, some commands are intended to be *alternates* for others, and it would be redundant to map all of them. For example, the **Plugin Parameter Page** section includes the commands **Previous Page**, **Next Page**, and **Increment Page**. In most situations, you would either map **Previous Page** and **Next Page** to a pair of buttons, *or* you would map

**Increment Page** to a knob – but you probably wouldn't map all three.

This flexibility allows you to lay out the commands in a way that makes the most sense for your MIDI controller. If your MIDI controller has more buttons than knobs (e.g. the X-Touch Mini), then you'd probably want to take the button approach. However, if your MIDI controller has more knobs than buttons (e.g. the MIDI Fighter Twister), then you might prefer the space-saving approach of using a single knob to switch parameter pages, rather than using up a pair of buttons for the same functionality.

A <u>complete list of all the Mappable Commands</u> is included in the Appendix of this manual.
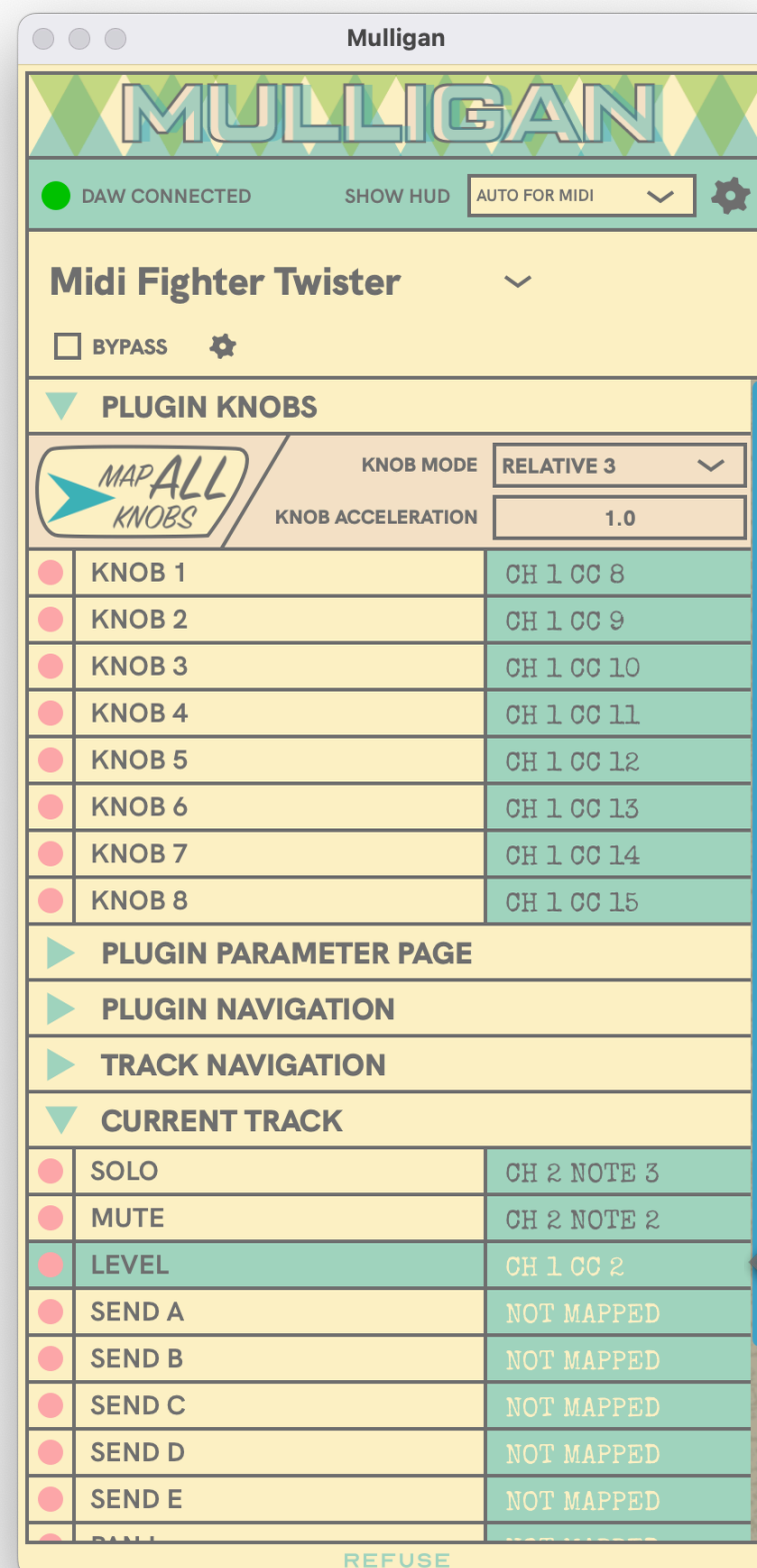
# Example: Mapping the Current Track Level
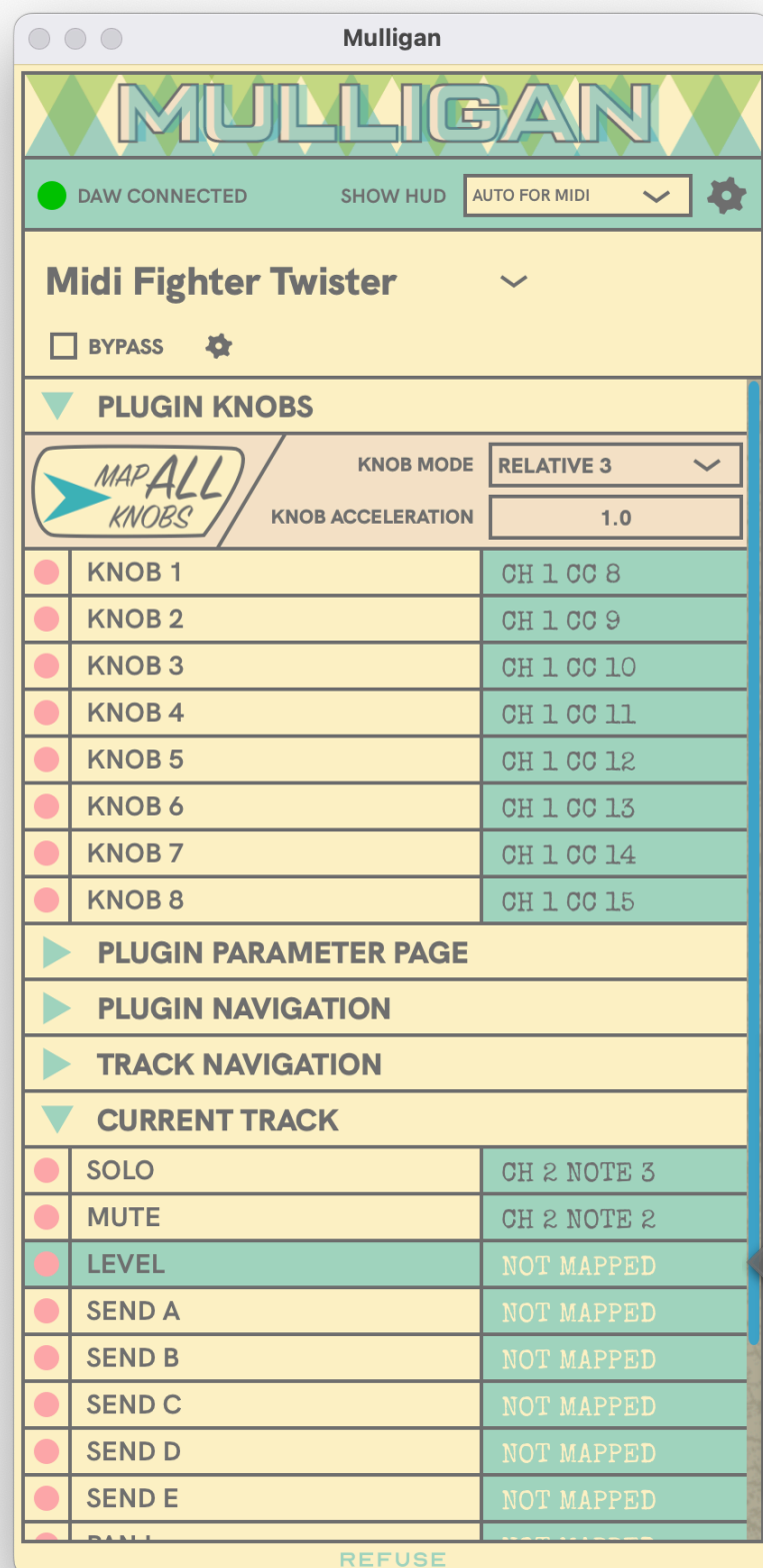


1. **Click** the **Level** command in the **Current Track** section to select it.

The **Mapping Assistant** window pops up to give additional information about the command.

2. **Turn** a knob **SLOWLY** to the left (counterclockwise).

The Mapping Assistant's status bar changes from **Ready To Map** to **Mapped**, and the knob you turned is now mapped to the Level command.
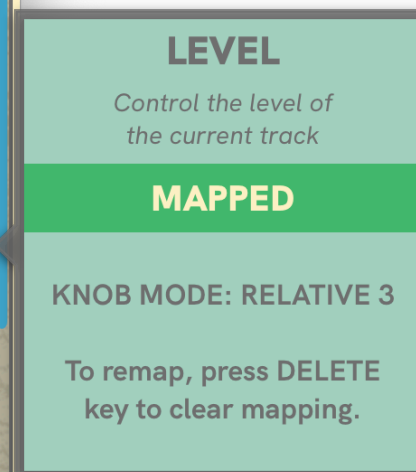
## Mapping Window Tips

- When mapping any of the knobs, you must turn them **slowly to the left (counterclockwise)**. That is the only way that Mulligan can **correctly detect which <u>knob mode</u>** your knobs are sending.

- The **Map All Knobs** button allows you to map the entire bank of 8 Plugin Knobs, just by turning the first one (again, **slowly to the left**). It works like magic, assuming that your 8 knobs send their MIDI CC messages using consecutive controller numbers.

- At the top of the **Plugin Knobs section** there are a few extra controls that affect only the Plugin Knob commands. The **Knob Mode menu** displays the knob mode detected for the bank of plugin knobs *only* – it has no effect on other commands. The **Knob Acceleration** slider appears only for **Relative** Plugin Knobs. If the knobs onscreen seem to be turning too slowly in response to the physical knob, try increasing the Knob Acceleration. It is a subjective matter of feel, and it won't break anything. The default is 1.0 (no acceleration).

- The **Soft Takeover** checkbox is available only for **Absolute** Plugin Knobs. When checked, Mulligan waits until the hardware knob's position matches that of the plugin's onscreen knob, before it starts affecting it.

- Double-clicking the MULLIGAN banner at the top of the window will shrink the window down to a minimal size. The top row of controls (the **DAW Connected** light and the **Show HUD** menu) will remain visible, for operation while working in the DAW, but the MIDI input and mapping controls will all be hidden. Double-clicking the banner a second time will return the window to its previous size.

- Jump in with a **Starter Pack** that we have provided for select MIDI controllers (see the Mulligan download for a complete list of the Starter Packs included). Created with input from experienced Mulligan users, a Starter Pack gives you an instant command layout optimized for your controller. Once you <u>import the mapping</u> from the Starter Pack, you are free to modify it like any other mapping, to adapt it to your own workflow.

# Optimizing Your Mappings

The basic mapping instructions covered thus far will get you up and running quickly with Mulligan. After a little time with it, however, you may wish to tweak your controller setup a bit further to optimize your mappings. The information in this section will help you to do that.

## Knob Mapping & Knob Modes

Mulligan uses knobs for many of its commands. In addition to the bank of eight **Plugin Knobs**, there are the **Increment Page** and **Increment Track** commands, the **Master Fader** command, and most of the **Current Track** section (except for the Solo and Mute buttons). Whenever setting a mapping for a knob, Mulligan needs to know what kind of data the knob is sending, in order to correctly set the **knob mode** for it.

That is why we emphasize that you must turn knobs **slowly to the left** during mapping – because that is the only way Mulligan can detect the correct knob mode.

There are two basic modes that knobs can use to send data: Absolute and Relative. The technical details about that difference are explained further in the **All About MIDI Controller Knobs** section of this manual.

As mentioned earlier, **we highly recommend using Relative knobs** (encoders sending incremental data) for almost all of Mulligan's knob commands*, instead of Absolute knobs.

If your MIDI controller uses **pots** (which have hard "stop" points at the ends of their rotation) for its knob hardware, then it can *only* send Absolute knob data, and Mulligan will translate that data internally (from Absolute to Relative) when necessary.

However, if you have a controller with **encoders** for knobs, yet Mulligan is still detecting the Absolute knob mode for it, then we recommend you investigate how to change it to send Relative knob data instead. See the Setting Up for Optimal Performance section for more details.

Please note that the **Knob Mode menu** at the top of the Plugin Knobs section displays the knob mode detected *for the bank of plugin knobs only*. For the commands in other sections that use knobs, the knob mode is detected individually per command, and is displayed in the Mapping Assistant window.

---

*\* The one exception to this rule is the Master Fader command, which is best mapped to an Absolute knob, or an actual fader.*

## Button Mapping

Some of Mulligan's commands are intended to be mapped to buttons, and it wouldn't really make sense to map them to knobs (for example, **Previous Page** and **Next Page**, or any of the transport commands).

For these types of commands, we recommend configuring your MIDI controller for a "Momentary" or "Hold" style button behavior (the terminology varies across manufacturers). Often, this is the default behavior, so you may not need to adjust anything. In Mulligan's <u>MIDI Monitor</u> you should see a Note or CC message with a value of 127 when the button is pressed down, followed by another message with a value of 0 when the button is released.

What you *don't* want is a "Toggle" button behavior, where alternate button presses send out different values (e.g. on first press, the button sends a value of 127, but then it requires a second press to send a value of 0).

## A Note About the Current Track Commands

Mulligan includes a set of commands that affect the currently focused track in a session. These include track **Level**, **Pan L**, **Pan R**, levels for **Sends A-E**, and **Pan Combo** (which combines Pan L and Pan R into a single knob).

The M-Audio Keyboard protocol was designed to work only with **encoders** for these track controls. In order to allow them to work with knobs that are pots or linear faders, Mulligan performs a special translation so that these types of controls can emulate encoders.

If possible, we recommend mapping the Current Track commands to Relative encoders, to avoid this translation.

If you *do* map Current Track commands to pots or linear faders, they will work, but their physical positions won't directly correspond to the positions of the faders and knobs onscreen. Moving the physical control will move the onscreen control immediately (without any "Soft Takeover" type of pick-up). This works well for making small adjustments, but you may run out of physical control range. If this should happen, the pot or fader can be "recalibrated" by sweeping it through its full range.

Also, please note that MIDI feedback is not sent out for Current Track commands, so LED rings on MIDI controllers won't show any values for them.

# Custom Plugin Mapping in Pro Tools

When you first try out a plugin under Mulligan control, there will be a default map in Pro Tools that determines how the plugin's parameters are mapped to the 8 plugin control knobs.

If a plugin has more than 8 parameters, there will also be multiple "pages" of parameters, that you can navigate through using Mulligan's **Plugin Parameter Page** commands.

Sometimes a plugin's default map will be fine as-is, but if not, Pro Tools allows you to change the order of the parameters by creating custom plugin maps. This is not directly a feature of Mulligan, but since many users have never seen these mapping features in Pro Tools before, we will point them out here.

Pro Tools displays its **Plug-In Map Controls** at the top of each plugin window **only** when a compatible control surface is connected (in this case, Mulligan). They look like this:



The **Map Options menu** (the little circle with the down arrow at the top) allows you to save, load, rename, delete, import, and export map presets. Multiple maps per plugin are allowed, and you can select which one should be the default using this menu.

The **Map Presets menu** (where it initially says "Factory Default Map") is where your custom plugin maps will show up after you create them.

## Create A Custom Plugin Map

To create a custom plugin map, start by clicking the **LEARN** button. Then, either click a plugin control onscreen, or make a selection from the **Parameter Menu** (located directly to the left of the Learn button). Now, turn a knob on your MIDI controller, and the mapping is established.

(If the plugin doesn't respond when you click controls onscreen to map them, please see this FAQ.)

Repeat the "click control then turn knob" process to include all the parameters you are interested in.

Use Mulligan's **Plugin Parameter Page** commands to navigate through multiple pages while mapping, and expand your custom map beyond a single page of 8 parameters.

When you are satisfied with your custom plugin map, click the **LEARN** button onscreen again to exit learn mode and resume normal operation.

After creating your custom plugin map, you might be surprised to see that the parameter pages from the Factory Default Map are still appearing (they will appear *after* any parameter pages you have added). In order to hide the default pages, check the **Hide Factory Default Pages** option in the Map Options menu.

One important caveat when making custom plugin mappings is that **mono and stereo instances of the same plugin use completely separate mappings** (and so does each multichannel format). Because of this, if you create a custom mapping using a mono instance of a plugin, then switch to a stereo instance, you won't see your custom map in the Map Presets menu. Yes, this is rather inconvenient. No, there's nothing we can do about it, as that functionality is all internal to Pro Tools.

To learn more about the plugin mapping facilities provided by Pro Tools, please see the *Pro Tools Reference Guide* section titled "**Plug-In Mapping**".

Custom plugin mapping is also covered as part of our Quick Start Tutorial video.

# The Heads-Up Display (HUD)

Mulligan's Heads-Up Display (HUD) is an overlay window that can automatically pop up at the bottom of the screen when adjusting plugins. Even when Pro Tools is the current application, the HUD window will float in front of it.

The HUD's purpose is to be a fast visual onscreen reference when you are adjusting knobs on your controller. It shows:
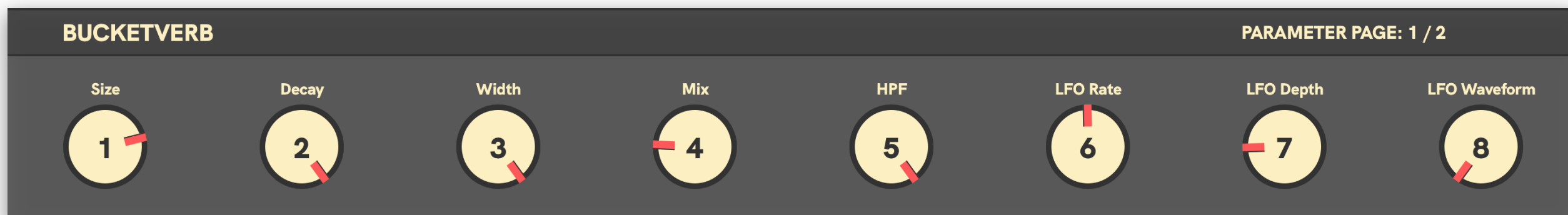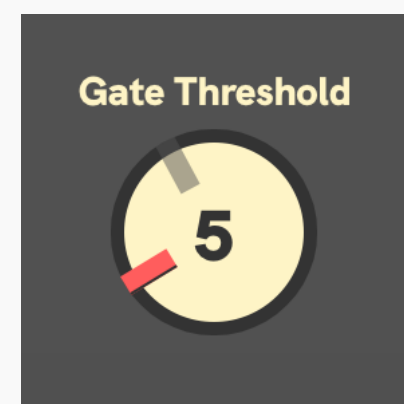
- The name of the plugin currently being controlled

- The 8 parameters of that plugin currently mapped to the 8 knobs of your MIDI controller

- The Parameter Page (bank of 8 parameters) that is active

Although the HUD displays parameter values on a dial, these are for reference only. You cannot turn the HUD dials with a mouse.

The only form of direct interaction with the HUD is to click on a plugin or parameter name, to allow you to **type in your own version of that name**. See the FAQ <u>Why do I see a mangled bunch of letters in the HUD where a plugin or parameter name should be?</u> for more details on that.

If you are using a controller with Absolute knobs, and have **Soft Takeover** engaged, the HUD also offers visual feedback to help you match up the knob position. It superimposes a "ghost" pointer (in grey) showing the hardware knob's position, with the regular pointer (in red) showing the parameter value:

**Gate Threshold**

**5**

---

**BUCKETVERB**                                         PARAMETER PAGE: 1 / 2

Size — 1    Decay — 2    Width — 3    Mix — 4    HPF — 5    LFO Rate — 6    LFO Depth — 7    LFO Waveform — 8

There are four **HUD display modes**, selectable from the **Show HUD** menu at the top of the Mapping Window:

1. **Off**
   *Never show the HUD.*

2. **Auto for MIDI**
   *Only show the HUD when controlling a mapped plugin using a MIDI controller.*

3. **Auto for All**
   *Same as **Auto for MIDI**, plus will also display the HUD when a mapped parameter changes from an onscreen control or DAW automation.*

4. **Keep On**
   *Always keep on, while the DAW connection is active.*

SHOW HUD    AUTO FOR MIDI    ⌄
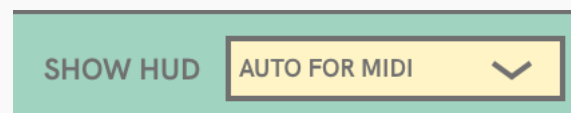
## Heads-Up Display (HUD) Tips

- Pressing and **holding** any of the **Plugin Navigation buttons** will keep the HUD showing for as long as the button is held down. This is helpful when first switching to a new plugin or a new plugin parameter page, to allow extra time to identify which knob is mapped to the parameter you'd like to change. Note that this feature only works if the Plugin Navigation buttons send a message (of value 0) upon button release.

- Initially, you may wish to keep Mulligan's HUD in the **Keep On** or **Auto for All** modes, as you get a feel for how the parameter mappings work. Eventually, however, the **Auto for MIDI** mode may feel the most intuitive, only showing the HUD when you are working hands-on with your MIDI controller.

## MIDI Device Controls

At the top of the Mapping Window are a few controls for working with your MIDI devices.
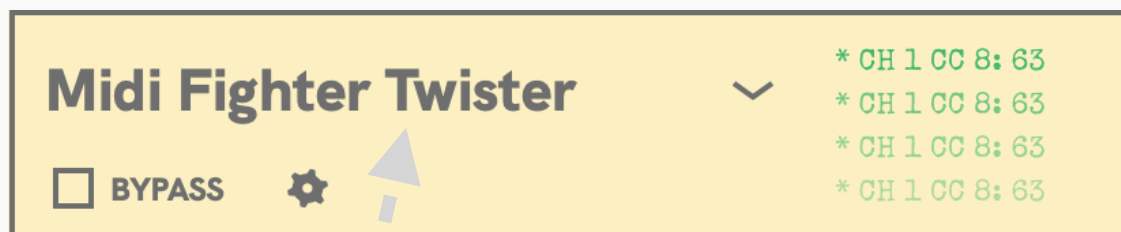


The **MIDI device menu** shows the name of the current MIDI device, and allows you to select a different MIDI device to focus on. **All connected MIDI devices are simultaneously active in Mulligan** (unless bypassed, see below), but the Mapping Window only displays mappings and controls for one device at a time.

The other controls described here affect only the currently focused MIDI device.

The **Bypass** checkbox allows you to suspend accepting input from the current MIDI device.

The **MIDI Monitor** on the right side shows recent MIDI messages received from the current device. When a MIDI message matches a command mapping, it will be shown in green. Unmapped messages are shown in red.

## Device Settings Menu

Accessed from the small gear icon under the MIDI device menu, the Device Settings Menu allows tweaking some behaviors for the current MIDI device.
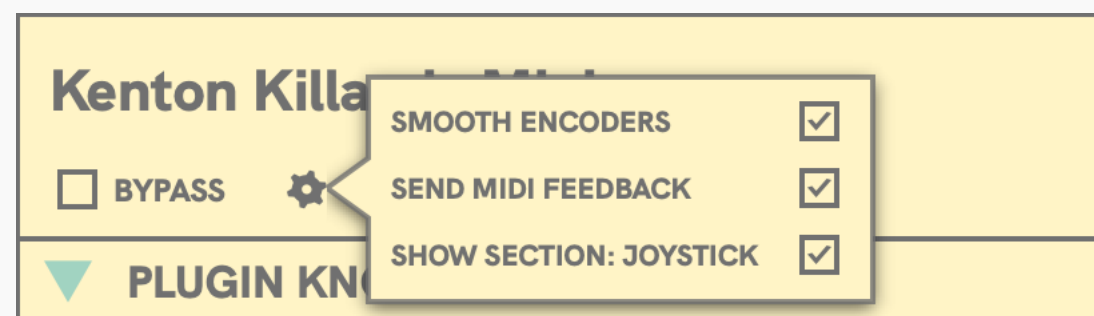


**Smooth Encoders** applies a filter to clean up incoming data from encoders sending underlined relative knob data. If your MIDI device has smooth (as in non-stepped) encoders for knobs, you can enable this option for more stable, smoother performance.

**Send Midi Feedback** defaults to "On", and you probably want to keep it on. It allows Mulligan to send MIDI to your device to provide visual feedback (e.g. LED rings around knobs). However, if you have a MIDI device that's acting strangely in response to the MIDI feedback, perhaps you would want to shut it off for troubleshooting.

**Show Section: Joystick** shows/hides the **Joystick / XY section** of commands. Since most devices *don't* have these types of controls, it defaults to "Off", to keep the Mapping Window clear of extra clutter.

# Enhanced Visual Feedback

Mulligan offers enhanced visual feedback for certain models of MIDI controller. The main purpose of this is to visually distinguish between when a knob is *unassigned* in the current parameter page, versus when a knob is *assigned* but its parameter is at its minimum value (e.g. zero or off).

How this might be visually represented depends on the particular model of controller, and it requires of course that the hardware have some kind of LEDs around its knobs.

Currently Mulligan detects the following MIDI controllers by name, and automatically sends extra messages to them for enhanced visual feedback:

## Midi Fighter Twister

When a knob is unassigned in the current parameter page, the multi-color (RGB) LED under the knob will be turned off. When a knob is assigned, the RGB LED under the knob will be lit (using the color set in the Midi Fighter Utility app).

In order for this to work, the *Switch MIDI Number* and the *Encoder MIDI Number* for each knob must be set to the same value. (They can have different *MIDI Channel* settings, however.)

## Kenton Killamix

When a knob is unassigned in the current parameter page, the knob's LED ring will have no segments lit. When a knob is assigned, but the parameter is set at its minimum value, the lowest segment of the LED ring will be lit.

## Behringer X-Touch Mini

When a knob is unassigned in the current parameter page, the knob's LED ring will have no segments lit. When a knob is assigned, but the parameter is set at its minimum value, the lowest segment of the LED ring will be lit.

In order for this to work, the Global Channel setting in the X-Touch Editor application must be left as channel 1 (the default).

## TouchOSC

Mulligan sends out plugin and parameter names in a custom SysEx format to enable displaying them in the TouchOSC control surface app. See the "TouchOSC iPad" Starter Pack included with the Mulligan download for details.

## User Data

### Saving

Mulligan automatically handles saving your user data, so you don't have to manually save anything. This includes your MIDI controller mappings, general application settings, and any custom plugin and parameter names you enter in the HUD.

A different MIDI map will be saved for each MIDI device you connect.

If you wish to backup or share your saved data, please use the **Export User Data** button in the **Settings Panel**. This will create a zip file archive of your settings files, and allow you to save that zip file wherever you choose.

If you need to restore from a backup, double-click the zip file to unzip the contents. Then the three XML files found within it should be placed into the folder:

```
~/Music/reFuse Software/Mulligan
```

You can use the **U** keyboard shortcut to automatically reveal this folder in the filesystem.

Please note that altering the XML files can render them unreadable by Mulligan, so editing them is not encouraged.

### Importing

Mulligan can import controller mappings from an XML file.

To import a mapping, use the **M** keyboard shortcut to open the import file dialog. From there, navigate to a `Mappings.xml` file that was generated by Mulligan.

Importing mapping data from a `Mappings.xml` file only imports the map for the **currently selected MIDI input device**. Any other saved maps (for other MIDI controllers) that you might have will be left untouched.

While the import does completely **replace** the current device map with the imported one, it also automatically makes a backup of your old settings, within a timestamped subfolder of the user data folder. This automatic backup ensures that, if you don't like the map you just imported, you could revert to your old one (use the **U** shortcut to find your timestamped backup folder, locate the `Mappings.xml` file within it, and then use the **M** shortcut again to re-import your old mapping from the file).

**Section 03**

# COMMANDS

## Commands List

The following commands are available for mapping in Mulligan. These descriptions also appear in the **Mapping Assistant** popover window when you click on a command.

| Command | Description |
| --- | --- |
| **Plugin Knobs** | |
| Knob 1 | Control plugin parameter 1 |
| Knob 2 | Control plugin parameter 2 |
| Knob 3 | Control plugin parameter 3 |
| Knob 4 | Control plugin parameter 4 |
| Knob 5 | Control plugin parameter 5 |
| Knob 6 | Control plugin parameter 6 |
| Knob 7 | Control plugin parameter 7 |
| Knob 8 | Control plugin parameter 8 |
| **Plugin Parameter Page** | |
| Previous Page | Move to previous page of plugin parameters |
| Next Page | Move to next page of plugin parameters |
| Increment Page | Use a knob to select parameter page |
| Page Return[1] | Reset page in Learn mode; jump b/t first & last pages |
| Page Bump[2] | While pressed, temporarily bumps up to next page |
| **Plugin Navigation** | |
| Next Plugin | Cycle through the plugins on a track |
| Show/Hide Plugin | Show/hide the active plugin window |
| Combo | A short tap triggers Next Plugin, while a long hold triggers Show/Hide Plugin |

| Track Navigation | |
| --- | --- |
| Previous Track | Move to previous track |
| Next Track | Move to next track |
| Increment Track (Knob) | Use a knob to select current track |
| **Current Track** | |
| Solo | Solo the current track |
| Mute | Mute the current track |
| Level | Control the level of the current track |
| Send A | Control the level of Send A on the current track |
| Send B | Control the level of Send B on the current track |
| Send C | Control the level of Send C on the current track |
| Send D | Control the level of Send D on the current track |
| Send E | Control the level of Send E on the current track |
| Pan L | Control the left (or mono) pan of the current track |
| Pan R | Control the right pan of the current track |
| Pan Combo | Control left and right panning with a single knob |
| **Miscellaneous** | |
| Master Fader | Control the rightmost Master Fader |
| Show/Hide HUD | Manually show/hide the HUD |
| **Transport** | |
| Stop | Stop the DAW transport |
| Play | Play the DAW transport |
| Stop/Play | Use a single button to Stop/Play the DAW transport |
| Rewind | Rewind the DAW transport |
| Fast Forward | Fast forward the DAW transport |
| Record | Arm the DAW for recording |
| Loop Play | Toggle Loop Playback on/off |

## Notes on Commands

### 1. Page Return

When switching in or out of plugin Learn mode in Pro Tools (i.e. when doing custom plugin mapping), Page Return will return you to the last parameter page you were on before entering Learn mode. (Pro Tools automatically jumps back to page 1 whenever entering or leaving Learn mode, which is not always desirable.)

Under normal operation (*not* involving Learn mode) Page Return will hop between the first and last parameter pages.

### 2. Page Bump

Page Bump provides a way to temporarily access the next parameter page of a plugin mapping, without taking your hand off the knob. **Map this command to the pushbutton function of your first plugin knob.** While you hold the knob down, you access parameter page +1, and when you release the knob, it reverts to the original page. When you design custom plugin mappings around this concept, **it's like adding another bank of virtual knobs to your controller**.

For example, you could map 4 bands of EQ across 8 knobs, as follows: each EQ band gets mapped to a pair of knobs. Start with Knob 1 mapped to the Low band frequency, and Knob 2 mapped to the Low band gain. Then, on the *next*

parameter page, map Knob 1 to Low band enable, and Knob 2 to Low band Q control.

Repeat this pattern for the other three EQ bands, assigning the knobs in pairs. So now the two parameter pages would look like this when viewed in the HUD:

| EQ3 7-BAND | | | | | | | PARAMETER PAGE: 1 / 2 |
|---|---|---|---|---|---|---|---|
| LOW FREQ | LOW GAIN | LOW-MID FREQ | LOW-MID GAIN | MID FREQ | MID GAIN | HIGH-MID FREQ | HIGH-MID GAIN |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

| EQ3 7-BAND | | | | | | | PARAMETER PAGE: 2 / 2 |
|---|---|---|---|---|---|---|---|
| LOW ENABLE | LOW Q | LOW-MID ENABLE | LOW-MID Q | MID ENABLE | MID Q | HIGH-MID ENABLE | HIGH-MID Q |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

Then in practice it would work like this: you could mostly use parameter page 1, where you control frequencies and gains for the four EQ bands. But for doing a quick tweak of the Q, or enabling/disabling the band, you would push down on a knob and engage Page Bump. That would put the Enable and Q parameters right there under your fingers, on the same knobs.

**!** Even though you only map the Page Bump command to the *first* plugin knob's pushbutton function, Mulligan will automatically detect when *any* of the eight plugin knobs are pushed (provided that the knob pushbuttons send consecutive note numbers), and use them to engage Page Bump as well.

## Optional Commands

### Joystick / XY

Mulligan 1.6 adds an optional section of commands for mapping input from joysticks (either free or center-sprung) and XY pads. These commands are specialized to deal with the idiosyncrasies of joystick and XY pad inputs, which by their nature behave differently than knobs. For the remainder of this section, we will refer to these types of inputs simply as "joystick" inputs, for brevity's sake.

The Joystick section will only be visible for a MIDI device when it is activated in the <u>Device Settings Menu.</u>

| ▼ JOYSTICK / XY | | | |
|---|---|---|---|
| 🔴 X1 | KNOB 1 | ⌄ | CH 1 CC 19 |
| 🔴 Y1 | KNOB 2 | ⌄ | CH 1 CC 20 |
| 🔴 X2 | NONE | ⌄ | NOT MAPPED |
| 🔴 Y2 | NONE | ⌄ | NOT MAPPED |

Commands are provided for two joysticks. X means the horizontal movement of the joystick, and Y means the vertical movement. So therefore, we have **X1** and **Y1** for the first joystick, and **X2** and **Y2** for the second (for those of you so lucky as to have a controller with dual joysticks!).

As you can see in the screenshot above, joystick commands display a popup menu in The Mapping Window, to select which command they actually trigger. These can be any of the eight **Plugin Knobs**, or any of the continuous controls found in the **Current Track** section.

While it was possible before to map a joystick directly to one of those other commands, it would have meant you couldn't simultaneously have a knob mapped to it.
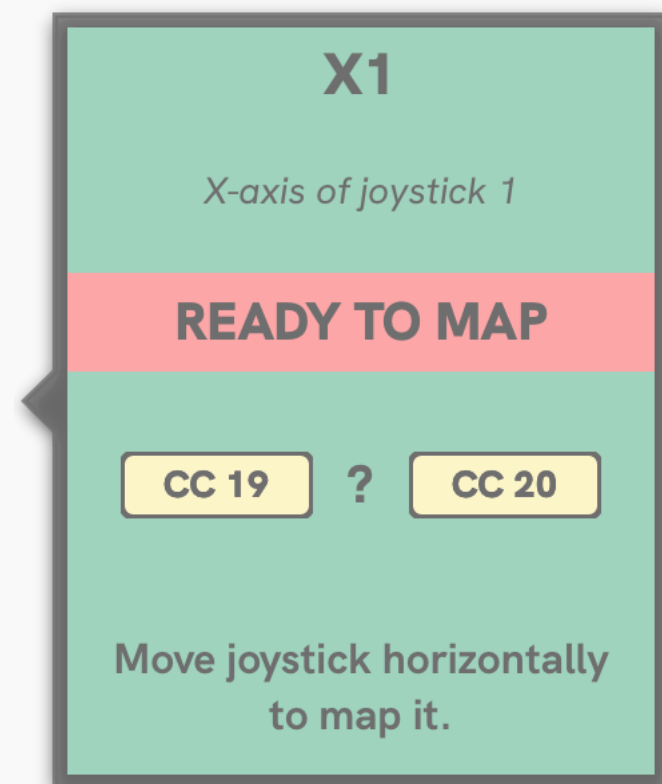
The idea with having a special joystick section is that you could have 8 knobs mapped to the 8 plugin controls per usual, and then *also* have a joystick mapped to 2 of those plugin controls at the same time. For some plugins (those with XY grids onscreen, or EQs with frequency/gain plots), joystick input feels intuitive and makes sense, but for other plugins it might feel sort of pointless. Now, with both knobs and joystick mapped, you can grab the joystick when it's a good match for a particular plugin, or just use knobs if it's not a good match.

In addition to this dual-mapping convenience, the joystick commands are optimized for handling joystick input data. One of the extra features they include is a mapping confirmation page in their Mapping Assistant windows. For knobs, the mapping process is pretty foolproof: click a command to map it, and turn the knob. However, with a
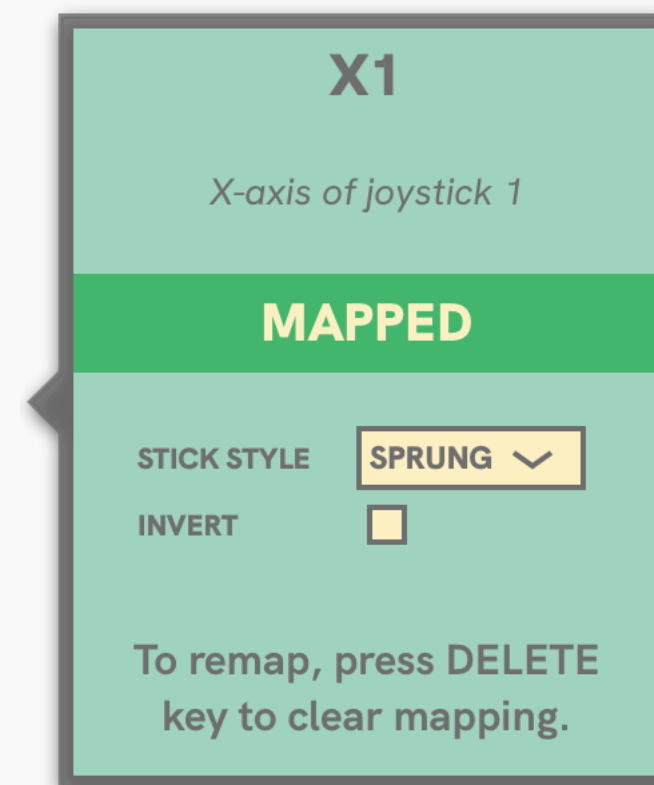
joystick, if you accidentally nudge the stick in the wrong direction at first, the mapping might grab the CC number for the Y-axis, when you wanted it to map to the X-axis. (In addition, some XY pads send X/Y values in pairs every time, making it impossible to map the correct axis, even if you had perfectly steady hands.)

So to make the mapping process for joysticks less frustrating, the Mapping Assistant will show you the two CC options that it detects. Then, by clicking on the one you want, you can complete the mapping:

**X1**

*X-axis of joystick 1*

**READY TO MAP**

| CC 19 | **?** | CC 20 |

**Move joystick horizontally to map it.**

Once the mapping is established, the Mapping Assistant will change to show the user-selectable options for that mapping:

**X1**

*X-axis of joystick 1*

**MAPPED**

STICK STYLE    SPRUNG ∨

INVERT    ☐

**To remap, press DELETE key to clear mapping.**

**Stick Style:** Choices are Free or Sprung (the latter is for return-to-center, sprung joysticks). If Sprung is selected, then values are changed when the stick is moved or held off-center.

**Invert:** Some joysticks may have an inverted axis (this is usually seen on the Y-axis) where they send zero at the top rather than at the bottom. This checkbox allows you to accommodate those cases.

26

Section 04

# FAQ

## Why do I see a mangled bunch of letters in the HUD where a plugin or parameter name should be?

What you're seeing there is the shortened (8 character) form of those names, just as they are sent to Mulligan from the DAW. It's not very pretty, and in some cases, it's downright nonsensical.

Mulligan attempts to substitute more readable versions of those garbled names for its display, but it does not have names for every plugin and parameter under the sun.

As a solution: You can click on the offending name in the HUD, and type in your own more readable substitution. This will be saved automatically with your user data.

## Why isn't the Learn button in the Pro Tools plugin window responding to clicking?

This seems to be a bug in Pro Tools. The workaround is when setting up the MIDI Controllers window, to make sure that Mulligan is in slot #1.

If you also use another controller that works best when in slot #1 (like a HUI device), then it may make sense to create custom plugin maps for all the plugins you regularly use, and then move Mulligan to another slot.

## I mapped a knob, but now it either stutters in place, or jumps between minimum and maximum, when I try to turn it. What went wrong?

These behaviors indicate that an incorrect knob mode was set for that command. Please repeat mapping the command in the Mapping Window, being sure to turn the knob slowly to the left so that Mulligan can detect the correct knob mode.

**I'm trying to make a custom plugin map in Pro Tools, but when I click on a plugin control, nothing happens. What do I do?**

There is a known issue with certain plugin brands, where their code is missing a required linkage between the plugin GUI and the plugin parameter system. Fortunately, there are a couple workarounds for it, which will allow you to create custom plugin maps regardless.
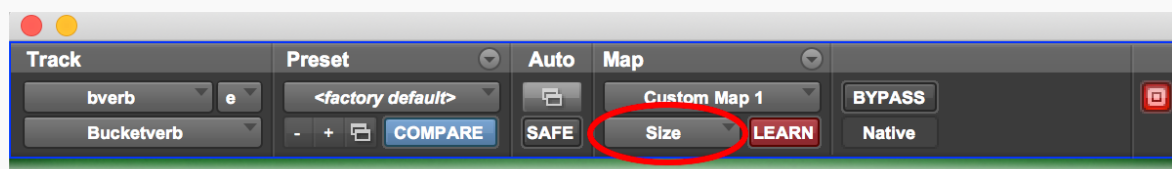


1. You can always use the **Parameter Menu** in the plugin window (the popup menu just to the left of the **LEARN** button) to select a parameter for mapping. The parameter menu defaults to reading "No Control". After clicking the Learn button, select the parameter you'd like to map from the menu, and twist a knob. Then pick another parameter from the menu, and twist a knob. This works just like the click & twist mapping method, except you're using the menu instead of clicking.

However, selecting the parameter from a menu is slower than just getting to click on it, and for some plugins with dozens (or even hundreds) of parameters, scrolling through that menu can be rather awkward. So, there is a second method, which *can* work, but again is dependent on the plugin manufacturer's implementation, so it's not guaranteed to work:

2. After clicking the **LEARN** button, hold down the "3-finger salute" keyboard shortcut and click on an onscreen control, aka `Command-Option-Control-click`. This may get the plugin to recognize that you're clicking on a control, even if it ignores plain clicks with no keyboard modifiers. (The shortcut of just `Command-Control-click` may also work for many plugins – so if the first one fails, try that one instead.)

Section 05

# APPENDIX

# All About MIDI Controller Knobs

When choosing a MIDI controller to use with Mulligan, the factor that can make the biggest difference to your experience is the type of knobs it has.

The bottom line is: **Mulligan works best with Encoders sending Relative knob data**. To understand why, we need to delve into knobs in detail.

## Pots vs Encoders

There are two kinds of knob hardware found on controllers:

- **Pots** have hard "stop" points at the ends of their rotation

- **Encoders** (aka *incremental knobs* or *rotary encoders*) can keep spinning endlessly in either direction

## Absolute Data vs Relative Data

There are two kinds of data that a knob can send:

- **Absolute** data describes the *position* of the knob, in the range from 0 to 127.

- **Relative** data describes the *change* in a knob when turned, with values like +1 (for a small clockwise turn), or -3 (for a larger counter-clockwise turn).

So how does **Pots vs Encoders** relate to **Absolute vs Relative**?

**Pots** are *only* capable of sending absolute knob data. However, most controllers with **encoders** have the ability to send either absolute *or* relative knob data. We'll call those two modes **absolute encoders** and **relative encoders** for short – but to be clear, the actual knob hardware is identical. The difference is just in which type of data the MIDI controller sends from the encoders.

For use with Mulligan, the hierarchy of preference for these knob types are:

**Relative Encoders > Absolute Encoders > Pots**

The reason for this is twofold.

First, why are encoders better than pots? With **encoders**, you never have to worry about a mismatch between the hardware knob and the onscreen knob. It's like the hardware knob automatically matches the position of the onscreen control.

In contrast, with **pots** you might have the hardware knob at 9 o'clock, but the onscreen knob is at 3 o'clock. If you turn the physical knob just a little bit, the onscreen knob will suddenly jump from 3 o'clock to 9 o'clock. Mulligan offers an

optional **Soft Takeover** mode to prevent these jumps when using pots, but it is faster and more intuitive to use an encoder.

Second, why are relative encoders more optimal than absolute encoders? Relative encoders do better with plugin controls that are on-off or multi-position **switches**.

Say you have a reverb plugin with a control for three different modes: *Plate*, *Hall*, and *Room*. With a **relative encoder**, it only takes a small turn to change modes. If you're currently on *Hall*, it's always one click to the right for *Room*, or one click to the left for *Plate*.

However, with an **absolute encoder**, it's a different story. The knob's sweep would be divided into three zones:



HALL

PLATE     ROOM
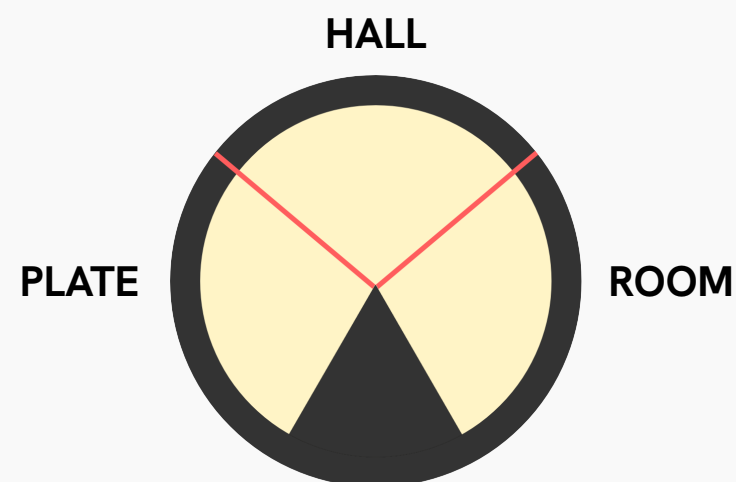
In order to switch between modes, you might have to turn the knob just a little (if it happens to be near one of those boundary lines) or a lot. This makes the knob action

unpredictable, and it can result in a lot of knob turning with no visible response, which does not feel intuitive in practice.

Even worse is using an absolute encoder with an on-off switch, because these are often divided into unequal zones. For example, an EQ *band enable* switch might have these two zones, where only a zero value turns it off:



ON

OFF
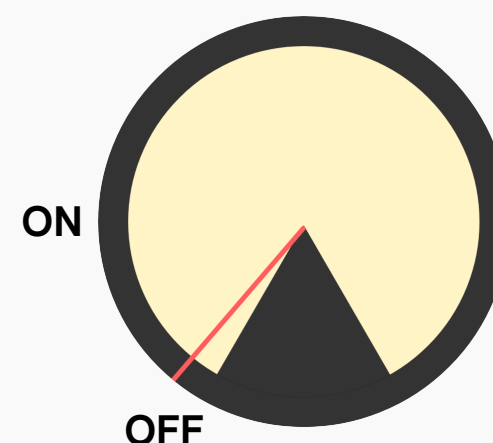
When you bring up this EQ plugin, if the control is in the On position, the absolute encoder will be turned up to its maximum. Therefore, in order to change it to Off, you'd have to sweep the control all the way down to the very bottom, which on an absolute encoder can require multiple spins of the knob. As before with the reverb mode example, this results in a lot of knob turning with no visible effect, instead of the instant response you would get from a relative encoder.

## Setting Up Knobs for Optimal Performance or: Beware the Absolute Encoder!

Given all the benefits we have shown to using relative encoders instead of absolute encoders, it is unfortunate that **most MIDI controllers with encoders send absolute data by default**.

For optimal performance with Mulligan, you must change your controller to send relative data instead. Depending on the make and model, this can be done by setting onboard options, or by using a software editor supplied by the manufacturer.

Further confusing the issue is the fact that there are multiple formats for sending relative encoder data, and that there is no consistent naming standard for those. In Mulligan, we have opted to name the relative knob modes as **Relative 1**, **Relative 2**, **Relative 3**, and **Relative 4**. Other companies have used technically descriptive names, like Signed Bit, Signed Bit 2, Bin Offset and Twos Complement. Still others have used rather cryptic looking names like **ENC 3FH/41H** (looking at you, MidiFighter Twister!).

Even among companies that use a simple numbered naming scheme like we do, one company's "Relative 1" might actually be a different format than another company's "Relative 1". So **you can't rely on matching the relative knob modes by name**.

Fortunately, Mulligan's automatic knob mode detection makes the naming a moot point. When setting up your MIDI controller, just pick the first relative mode available, and make sure that it's the same for all the encoders.

Then, when mapping your controller in Mulligan, turn the knobs *slowly* to the left. This allows Mulligan to detect which knob mode is in use. Don't worry if the knob mode name in Mulligan doesn't match what you had selected in your MIDI controller setup – if it works right, it *is* right.

## Keyboard Shortcuts

The following keyboard shortcuts are available in Mulligan, by pressing just the key indicated (with no modifier keys).

| Action | Shortcut |
| --- | --- |
| Toggle transport stop/play | Spacebar |
| Clear selected mapping | Delete |
| Clear all mappings for current device | Option-Delete |
| **HUD** window manual show/hide | H |
| Mapping Window **Grow**/Shrink | G |
| Show **Authorization** dialog | A |
| Copy diagnostic **Info** to clipboard | I |
| Show **User** Data folder in Finder | U |
| Import **Map** for current device | M |
| Show Settings Panel | , |
| Close current popup window | Esc |
| Show Visual Guide help window | ? |

reFuse Software, LLC
Portland, OR, USA
www.refusesoftware.com