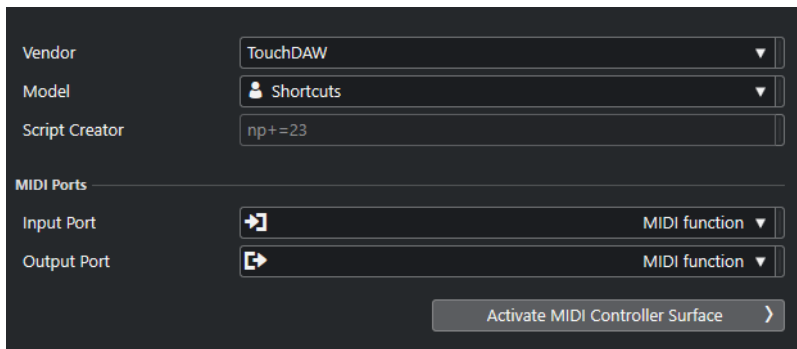


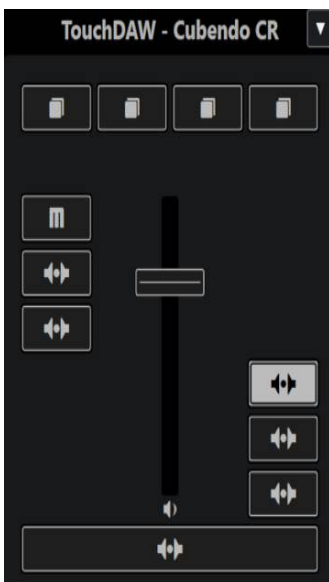
## A collection of Cubase / Nuendo MIDI remote script samples for TouchDAW (v.2.3.1+).

For installation open the 'MIDI Remote' tab in the DAW's lower zone and click the large '+' button. On the upper left-hand side click 'Import Script' and browse for the .midiremote file to import.

The scripts do not rely on specific MIDI ports. After importing you will see something like this:



Select the desired ports in the lower half, click 'Activate MIDI Controller Surface' and you will get a UI representation of the controller.



**'Cubendo CR.midiremote'** – Basic control room remote with four subpages for main, phones and two cue mixes each with the usual mute, dim and metronome toggles, plus monitor selection and a large talkback button. To be used with the equally named Workshop preset.



**'Shortcuts.midiremote'** - A basic 'single touch key command' utility that goes with the 'Pad Grid' preset on the Workshop screen.

Besides eventually being useful, it is meant to demo dynamic labelling and coloring in the app.

Set up with four subpages of which only two have pre-assigned commands. To add and edit shortcuts you need to work on the script, because Cubase's Mapping Assistant can't handle subpages (if a single page is good enough for you, it will do).



**'MIDI Mode IQC.midiremote'**  
- Controller for the MIDI Mode option on the app's main screen.

Premapped to control 'Focused Quick Controls' on Instrument channels via the faders, open instrument and edit windows, select channels and set them to automation write mode via the buttons. Rotaries are organized in subpages and control volume, pan and send levels. The master fader is linked to the main monitor level via Cubase's Control Room.

**'MIDI Mode.midiremote'** - Unmapped version of the former, only defining the UI representation. Use Cubase's Mapping Assistant to link the controls to functions of your choice.



**'AI Fader.midiremote'** – Emulates the 'AI Knob' on the discontinued CC121 hardware controller. To be used with the equally named Workshop preset. Controls the 'value under mouse pointer' while also displaying the name of the edited component, the parameter's name and its current value.

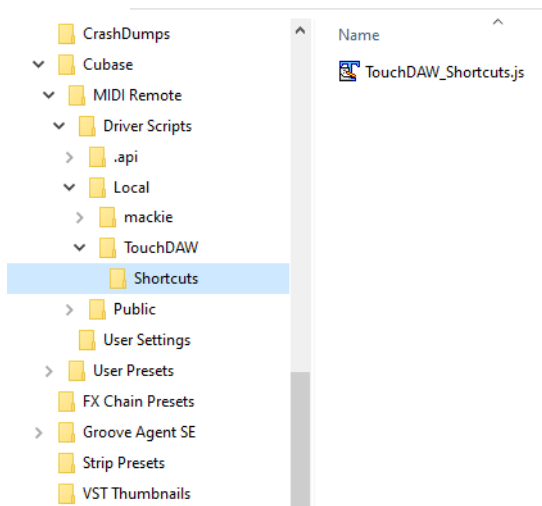
The midiremote APIs do not yet feature Fader-Touch functionality. The script uses a separate button not present in the app's UI to hook up the Lock function. That button is then linked to the MIDI generated when touching / releasing the fader in the app.

=====

After importing the .midiremote files you will find uncompressed javascript files at

Windows: 'C:\Users\USERNAME\Documents\Steinberg\Cubase\MIDI Remote\Driver Scripts\Local\TouchDAW'.

Mac: 'Users/USERNAME/Documents/Steinberg/Cubase/MIDI Remote/Driver Scripts/Local/TouchDAW'



These contain additional comments and can be edited in-place if you want to.

For further details, API documentation, how to use the 'Mapping Assistant' etc. please refer to [Steinberg's documentation](#).

=> These scripts are meant to be simple examples. They will likely not exactly match your setup or needs and will need adaption.

All scripts rely on TouchDAW's default Note & CC assignments in the associated presets, resp. the app's MIDI mode settings. If you had edited any of those, parts of the controllers may not work as intended.

#### Known issues:

When opening a project or refreshing controller scripts Cubase will briefly close and reopen a remote's MIDI ports. This is a problem with RTP MIDI as it will break the external network connections. You will need to reconnect in Audio MIDI Setup / rtpMIDI or wait a short moment for automatic reconnections if you use MIDIHub or have TouchDAW initiate sessions. The biggest issue with this however is that it effectively makes it impossible to initiate the controller at some defined point (like in `deviceDriver.onActivate`). I am not sure why Steinberg think it's necessary to toggle the ports off and on again and can only hope that this will be reverted at some point.

MIDI remotes 'consume' the MIDI data they are set to handle. If you use the app's keyboard or pads and have a MIDI Remote hooked up to the same MIDI port, some notes may get swallowed by the remote. Use a MIDI channel not used by the remote on the keyboard or move the remote's notes out of range of what you will need for musical input.

The whole MIDI remote system does not seem to be fully stable to me so far. There are points where Cubase will just stop listening to a controller and won't send updates any longer. This may happen when adding instrument tracks or performing other heavy weight actions. Normally it starts working again after poking around in Cubase's UI a bit (like selecting channels or opening instrument windows).