16.5. Mixxx Controls

Nearly every knob, button, or fader you see in Mixxx's interface is controllable via Mixxx's "control" system. The control system allows skins, MIDI controllers, HID controllers and keyboards to control Mixxx via a single interface.

A control is identified by a "group" (which is used for grouping associated controls) and a "key" (the name of the individual control).

For example, the volume fader for Deck 1 is identified by the group `[Channel1]` and key `volume`. Similarly, the volume fader for Sampler 1 is identified by the group `[Sampler1]` and key `volume`.

The group is used to collect all the controls that affect one component of Mixxx into one collection. Some groups have a high overlap of controls in common (e.g. samplers, decks, and the preview deck all share the same control keys).

In addition to controlling Mixxx, the control system can be used to inspect Mixxx's state. For example, the sample rate of the track loaded in Deck 1 can be accessed via the `[Channel1],track_samplerate` control. You can read the `[Channel3],play` control to determine whether Deck 3 is playing.

The default value range is 0.0 to 1.0, unless otherwise noted. Binary means that it is either 'ON' (non-zero) or 'OFF' (zero).

Hint

Discovering Controls used in Skins

You can view the control connected to any part of a skin by running Mixxx with the `--developer` command line option and hovering your mouse cursor over part of the skin. If no tooltip appears, enable tooltips for the Library and Skin in Options ‣ Preferences ‣ Interface .

Hint

Changing any control from the GUI in Developer Mode

When running Mixxx in Developer Mode (with the `--developer` command line option), you can view and manually set the state of any control in Mixxx by going to Developer ▸ Developer Tools .

Hint

Simplify mapping of more complex behaviour

While simple mappings with just a few buttons, knobs and LEDs can easily be created with the MIDI Wizard and some basic scripting, implementing more complex behaviour like switching deck layers or pad grid modes can be tedious and error-prone. For these cases you can use Mixxx' Comonents-JS library which provides building blocks for single controls as well as entire containers like decks and effect units.

See also

See Control Index for a full list.

ControlPotMeter controls

The following extensions add some features to `ControlPotMeter` controls (volume, crossfader, ...). Use in conjunction with `[ChannelN]`, `[SamplerN]`, `[Master]`, ... groups.

Control Suffix	Description, example
`_up`	Increases the value, e.g. `[ChannelN],rate_perm_up` sets the speed one step higher (4 % default)
`_down`	Decreases the value, sets the speed one step lower (4 % default)
`_up_small`	Increases the value by smaller step, sets the speed one small step higher (1 % default)
`_down_small`	Decreases the value by smaller step, sets the speed one small step lower (1 % default)
`_set_one`	Sets the value to 1.0, sets the channel volume to full
`_set_minus_one`	Sets the value to -1.0, sets the channel volume to zero
`_set_default`	Input: sets the control to its default, return to default waveform zoom level
`_set_default`	Output: set to 1.0 if the control is at its default, light up the pitch fader center indicator
`_set_zero`	Sets the value to 0.0, put the crossfader in the middle again
`_toggle`	Sets the value to 0.0 if the value was > 0.0, and to 1.0 if the value was 0.0, will cut off/on a track while you're playing
`_minus_toggle`	Sets the value to -1.0 if the value was > -1.0, and to 1.0 if the value was -1.0, can tilt the crossfader from left to right

These controls can be used in JavaScript files like this:

```
// This won't work:
engine.setValue(group, "pitch_up_small", 1.0);

// Use this instead:
script.triggerControl(group, "pitch_up_small", 50);
```

To use `*_toggle` the respective shortcut for scripts is:

```
script.toggleControl(group, "keylock_toggle", 100);
```

The `[App]` group

The `[App]` group contains controls that do not belong to a specific channel, the mixer or the effects engine.

[App] `indicator_250ms`

Alternates between 0.0 and 1.0 every 250 milliseconds.

This control may be used to implement a blinking LED in JavaScript and is guaranteed to light up at the same time as `[ChannelN],cue_indicator` and `[ChannelN],play_indicator` when these are blinking (depending on the currently chosen cue mode).

Range:	binary, read-only
Feedback:	None

New in version 2.4.0.

[App] `indicator_500ms`

Alternates between 0.0 and 1.0 every 500 milliseconds.

This control may be used to implement a blinking LED in JavaScript and is guaranteed to light up at the same time as `[ChannelN],cue_indicator` and `[ChannelN],play_indicator` when these are blinking (depending on the currently chosen cue mode).

Range:	binary, read-only
Feedback:	None

New in version 2.4.0.

[App] `num_decks`

The number of decks currently enabled.

Range:	integer
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],num_decks` control.

[App] `num_samplers`

The number of samplers currently enabled.

Range:	integer
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],num_samplers` control.

[App] `num_preview_decks`

The number of preview decks currently enabled.

Range:	integer
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],num_preview_decks` control.

[App] `num_microphones`

The number of microphone inputs that can be configured.

Range:	integer
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],num_microphones` control.

[App] `num_auxiliaries`

The number of auxiliary inputs that can be configured.

Range:	integer
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],num_auxiliaries` control.

[App] `samplerate`

The current output sample rate (default: 44100 Hz).

Range:	absolute value (in Hz)
Feedback:	None

New in version 2.4.0: Replaces the deprecated `[Master],samplerate` control.

The `[Master]` group

The `[Master]` group generally corresponds to controls that affect the mixing engine. This will bear some similarity to what you will find on a DJ mixer (e.g. crossfader controls, headphone cueing controls, etc.).

[Master] `audio_latency_usage`

Reflects fraction of latency, given by the audio buffer size, spend for audio processing inside Mixxx. At value near 25 % there is a high risk of buffer underflows

This is a ControlPotMeter control.

Range:	0 .. 25 %
Feedback:	latency meter

New in version 2.0.0.

[Master] `audio_latency_overload`

Indicates a buffer under or over-flow. Resets after 500 ms

This is a ControlPotMeter control.

Range:	binary
Feedback:	Overload indicator

New in version 2.0.0.

[Master] `audio_latency_overload_count`

Counts buffer over and under-flows. Max one per 500 ms

Range:	0 .. n
Feedback:	Counter in hardware preferences

New in version 2.0.0.

[Master] `balance`

Adjusts the left/right channel balance on the Main output.

This is a ControlPotMeter control.

Range:	-1.0..1.0
Feedback:	Center Balance knob

[Master] `booth_enabled`

Indicates whether a Booth output is configured in the Sound Hardware Preferences.

Range:	binary
Feedback:	Booth gain knob shown or hidden

New in version 2.1.0.

[Master] `booth_gain`

Adjusts the gain of the Booth output.

This is a ControlPotMeter control.

Range:	0.0...1.0...5.0
Feedback:	Booth gain knob

New in version 2.1.0.

[Master] `crossfader`

Adjusts the crossfader between players/decks (-1.0 is all the way left).

This is a ControlPotMeter control.

Range:	-1.0..1.0
Feedback:	Crossfader slider

[Master] `crossfader_down`

Moves the crossfader left by 1/10th.

Range:	binary
Feedback:	Crossfader slider

[Master] `crossfader_down_small`

Moves the crossfader left by 1/100th.

Range:	binary
Feedback:	Crossfader slider

New in version 1.10.0.

[Master] `crossfader_up`

Moves the crossfader right by 1/10th.

Range:	binary
Feedback:	Crossfader slider

[Master] `crossfader_up_small`

Moves the crossfader right by 1/100th.

Range:	binary
Feedback:	Crossfader slider

New in version 1.10.0.

[Master] `duckStrength`

Microphone ducking strength

This is a ControlPotMeter control.

Range:	0.0..1.0
Feedback:	Strength knob

New in version 2.0.0.

[Master] `enabled`

Indicator that the main mix is processed.

Range:	binary
Feedback:	None

New in version 2.0.0.

[Master] `gain`

Adjusts the gain for the main output as well as recording and broadcasting signal.

This is a ControlPotMeter control.

Range:	0.0..1.0..5.0
Feedback:	Main volume knob

New in version 2.0.0.

[Master] `headEnabled`

Indicator that the headphone mix is processed.

Range:	binary
Feedback:	None

New in version 2.0.0.

[Master] `headGain`

Adjusts the headphone output gain.

This is a ControlPotMeter control.

Range:	0.0..1.0..5.0
Feedback:	Headphone volume knob

New in version 2.0.0.

[Master] `headMix`

Adjusts the cue/main mix in the headphone output.

This is a ControlPotMeter control.

Range:	default
Feedback:	Pre/Main knob

[Master] `headSplit`

Splits headphone stereo cueing into right (main mono) and left (PFL mono).

Range:	binary
Feedback:	Split Cue button

New in version 2.0.0.

[Master] `latency`

Latency setting (sound buffer size) in milliseconds (default 64).

Range:	>=0 (absolute value)
Feedback:	Latency slider in the prefs

[Master] `num_effectsavailable`

The number of available effects that can be selected in an effect slot.

Range:	integer, read-only
Feedback:	None

New in version 2.1.0.

[Master] `PeakIndicator`

Indicates when the signal is clipping (too loud for the hardware and is being distorted) (composite).

This is a ControlPotMeter control.

Range:	binary

Feedback:	Clip light (mono)

[Master] `PeakIndicatorL`

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the left channel.

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (left)

[Master] `PeakIndicatorR`

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the right channel.

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (right)

[Master] `talkoverDucking`

Toggle microphone ducking mode (OFF, AUTO, MANUAL)

Range:	FIXME
Feedback:	Ducking mode button

New in version 2.0.0.

[Master] `VuMeter`

Outputs the current instantaneous main volume (composite).

This is a ControlPotMeter control.

Range:	default
Feedback:	Main meter (mono)

[Master] `VuMeterL`

Outputs the current instantaneous main volume for the left channel.

This is a ControlPotMeter control.

Range:	default
Feedback:	Main meter L

[Master] `VuMeterR`

Outputs the current instantaneous main volume for the right channel.

This is a ControlPotMeter control.

Range:	default
Feedback:	Main meter R

Decks, Preview Decks and Samplers

Each deck in Mixxx corresponds to a `[ChannelN]` group. Whenever you see `[ChannelN]`, think "Deck N". N can range from 1 to the number of active decks in Mixxx.

Preview decks and Sample decks ("samplers") in Mixxx are identical to regular decks, they simply have a different purpose (previewing tracks or playing samples, respectively). Any control listed above for `[ChannelN]` will work for a samplers and preview decks, just replace `[ChannelN]` with `[PreviewDeckN]` or `[SamplerN]`.

See also

There are some additional global controls for samplers.

[ChannelN] `back`

[PreviewDeckN] `back`

[SamplerN] `back`

Fast rewind (REW)

Range:	binary
Feedback:	<< button

[ChannelN] `bpmlock`

[PreviewDeckN] `bpmlock`

[SamplerN] `bpmlock`

Toggle the beatgrid/BPM lock state.

Range:	binary
Feedback:	The lock icon of the track is activated/deactivated.

New in version 2.5.0.

[ChannelN] `beat_active`

[PreviewDeckN] `beat_active`

[SamplerN] `beat_active`

Indicates, depending on the play direction, how the player is currently positioned to the closest beat.

An LED controlled by beat_active can be used for beat matching or for finding a beat using jog or control vinyl.

Note

In case of fractional loops (e.g. 1/32), the rate of beat events can be very high. You should test if your controller is capable to process this update rate. If not, inhibt the beat indication for short loops, depending on the value of *[ChannelN],beatloop_size*.

Value	Play direction	Position
0		Set when play direction changes or +-20% of the distance to the previous/next beat is reached
1	Forward	Set at a beat
2	Reverse	Set at a beat

Range: real number, read-only

Feedback: None

New in version 1.10.0: (Reverse indication added in 2.4.0)

[ChannelN] **`beat_closest`**

[PreviewDeckN] **`beat_closest`**

[SamplerN] **`beat_closest`**

Its value is set to the sample position of the closest beat of the active beat and is used for updating the beat LEDs.

Range: -1, 0.0, real-valued

Feedback: None

[ChannelN] **`beat_distance`**

[PreviewDeckN] **`beat_distance`**

[SamplerN] **`beat_distance`**

Outputs the relative position of the play marker in the section between the the previous and next beat marker.

Range: 0.0 - 1.0, real-valued

Feedback: None

[ChannelN] **`beatjump`**

[PreviewDeckN] **`beatjump`**

[SamplerN] **`beatjump`**

Jump forward (positive) or backward (negative) by N beats. If a loop is active, the loop is moved by X beats.

Range: any real number within the range, see `[ChannelN],beatloop_X_activate`
Feedback: Player jumps forward or backward by X beats.

New in version 2.0.0.

[ChannelN] **`beatjump_size`**

[PreviewDeckN] **`beatjump_size`**

[SamplerN] **`beatjump_size`**

Set the number of beats to jump with `beatjump_forward` / `beatjump_backward`.

Range: positive real number
Feedback: Beatjump size spinbox

New in version 2.1.0.

[ChannelN] **`beatjump_size_halve`**

[PreviewDeckN] **`beatjump_size_halve`**

[SamplerN] **`beatjump_size_halve`**

Halve the value of `beatjump_size`.

Range: binary
Feedback: Beatjump size spinbox

New in version 2.4.0.

[ChannelN] **`beatjump_size_double`**

[PreviewDeckN] **`beatjump_size_double`**

[SamplerN] **`beatjump_size_double`**

Double the value of `beatjump_size`.

Range: binary
Feedback: Beatjump size spinbox

New in version 2.4.0.

[ChannelN] **`beatjump_forward`**

[PreviewDeckN] **`beatjump_forward`**

[SamplerN] **`beatjump_forward`**

Jump forward by `beatjump_size`. If a loop is active, the loop is moved forward by X beats.

Range: binary

Feedback: Player jumps forward by `beatjump_size`.

New in version 2.1.0.

[ChannelN] **`beatjump_backward`**

[PreviewDeckN] **`beatjump_backward`**

[SamplerN] **`beatjump_backward`**

Jump backward by `beatjump_size`. If a loop is active, the loop is moved backward by X beats.

Range: binary

Feedback: Player jumps backward by `beatjump_size`.

New in version 2.1.0.

[ChannelN] **`beatjump_X_forward`**

[PreviewDeckN] **`beatjump_X_forward`**

[SamplerN] **`beatjump_X_forward`**

Jump forward by X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512. If a loop is active, the loop is moved forward by X beats.

Range: binary

Feedback: Player jumps forward by X beats.

New in version 2.0.0.

[ChannelN] **`beatjump_X_backward`**

[PreviewDeckN] **`beatjump_X_backward`**

[SamplerN] **`beatjump_X_backward`**

Jump backward by X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512. If a loop is active, the loop is moved backward by X beats.

Range:	binary
Feedback:	Player jumps backward by X beats.

New in version 2.0.0.

[ChannelN] **`beatloop_activate`**

[PreviewDeckN] **`beatloop_activate`**

[SamplerN] **`beatloop_activate`**

Set a loop that is `beatloop_size` beats long and enables the loop

Range:	binary
Feedback:	A loop is shown over `beatloop_size` beats

New in version 2.1.0.

[ChannelN] **`beatloop_X_activate`**

[PreviewDeckN] **`beatloop_X_activate`**

[SamplerN] **`beatloop_X_activate`**

Activates a loop over X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512

Range:	binary
Feedback:	A loop is shown over X beats.

New in version 1.10.0.

[ChannelN] **`beatloop_size`**

[PreviewDeckN] **`beatloop_size`**

[SamplerN] **`beatloop_size`**

Set the length of the loop in beats that will get set with `beatloop_activate` and `beatlooproll_activate`. Changing this will resize an existing loop if the length of the loop matches `beatloop_size`.

Range:	positive real number
Feedback:	Beatloop size spinbox and possibly loop section on waveform

New in version 2.1.0.

[ChannelN] **`beatloop_X_toggle`**

[PreviewDeckN] `beatloop_X_toggle`

[SamplerN] `beatloop_X_toggle`

Toggles a loop over X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512

Range:	binary
Feedback:	A loop is shown over X beats.

New in version 1.10.0.

[ChannelN] `beatloop_X_enabled`

[PreviewDeckN] `beatloop_X_enabled`

[SamplerN] `beatloop_X_enabled`

1 if beatloop X is enabled, 0 if not.

Range:	binary
Feedback:	Beatloop X button in skin is lit.

New in version 1.10.0.

[ChannelN] `beatlooproll_activate`

[PreviewDeckN] `beatlooproll_activate`

[SamplerN] `beatlooproll_activate`

Activates a rolling loop over `beatloop_size` beats. Once disabled, playback will resume where the track would have been if it had not entered the loop.

Range:	binary
Feedback:	A loop overlay is shown over `beatloop_size` beats on waveform.

New in version 2.1.0.

[ChannelN] `beatlooproll_X_activate`

[PreviewDeckN] `beatlooproll_X_activate`

[SamplerN] `beatlooproll_X_activate`

Activates a rolling loop over X beats. Once disabled, playback will resume where the track would have been if it had not entered the loop. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512

Range:	binary

Feedback:	Beatloop X button in skin is lit. A loop overlay is shown over X beats on waveform.

New in version 1.11.0.

[ChannelN] **`beats_adjust_faster`**

[PreviewDeckN] **`beats_adjust_faster`**

[SamplerN] **`beats_adjust_faster`**

Adjust the average BPM up by +0.01

Range:	binary
Feedback:	The beatgrid lines move closer to each other.

New in version 2.0.0.

[ChannelN] **`beats_adjust_slower`**

[PreviewDeckN] **`beats_adjust_slower`**

[SamplerN] **`beats_adjust_slower`**

Adjust the average BPM down by -0.01.

Range:	binary
Feedback:	The beatgrid lines move further apart from each other.

New in version 2.0.0.

[ChannelN] **`beats_translate_curpos`**

[PreviewDeckN] **`beats_translate_curpos`**

[SamplerN] **`beats_translate_curpos`**

Adjust beatgrid so closest beat is aligned with the current playposition.

Range:	binary
Feedback:	The beatgrid moves to align with current playposition.

New in version 1.10.0.

[ChannelN] **`beats_translate_match_alignment`**

[PreviewDeckN] **`beats_translate_match_alignment`**

[SamplerN] **`beats_translate_match_alignment`**

Adjust beatgrid to match another playing deck.

Range: binary
Feedback: Instead of syncing the beatgrid to the current playposition, sync the beatgrid so the nearest beat lines up with the other track's nearest beat.

New in version 2.0.0.

[ChannelN] `beats_translate_earlier`

[PreviewDeckN] `beats_translate_earlier`

[SamplerN] `beats_translate_earlier`

Move beatgrid to an earlier position.

Range: binary
Feedback: The beatgrid moves left by a small amount.

New in version 2.0.0.

[ChannelN] `beats_translate_later`

[PreviewDeckN] `beats_translate_later`

[SamplerN] `beats_translate_later`

Move beatgrid to a later position.

Range: binary
Feedback: The beatgrid moves right by a small amount.

New in version 2.0.0.

[ChannelN] `shift_cues_earlier`

[PreviewDeckN] `shift_cues_earlier`

[SamplerN] `shift_cues_earlier`

Range: binary
Feedback: All cue markers move left by 10ms.

New in version 2.3.0.

[ChannelN] `shift_cues_later`

[PreviewDeckN] `shift_cues_later`

[SamplerN] `shift_cues_later`

Range: binary
Feedback: All cue markers move right by 10ms.

New in version 2.3.0.

[ChannelN] **`shift_cues_earlier_small`**

[PreviewDeckN] **`shift_cues_earlier_small`**

[SamplerN] **`shift_cues_earlier_small`**

Range:	binary
Feedback:	All cue markers move left by 1ms.

New in version 2.3.0.

[ChannelN] **`shift_cues_later_small`**

[PreviewDeckN] **`shift_cues_later_small`**

[SamplerN] **`shift_cues_later_small`**

Range:	binary
Feedback:	All cue markers move right by 1ms.

New in version 2.3.0.

[ChannelN] **`beats_undo_adjustment`**

[PreviewDeckN] **`beats_undo_adjustment`**

[SamplerN] **`beats_undo_adjustment`**

Restores the beatgrid state before the last beatgrid adjustment done with the above *beats_* controls.

The undo stack holds up to ten beatgrid states. For changes done in quick succession (less than 800 milliseconds between actions), e.g. repeated `beats_translate_earlier`, only the first state is stored.

Range:	binary
Feedback:	The beatgrid is restored.

New in version 2.5.0.

[ChannelN] **`beatsync`**

[PreviewDeckN] **`beatsync`**

[SamplerN] **`beatsync`**

Syncs the tempo and phase (depending on quantize) to that of the other track (if BPM is detected on both).

range: binary

feedback: The Sync button flashes and the tempo slider snaps to the appropriate value.

Changed in version 1.10.0.

[ChannelN] `beatsync_phase`

[PreviewDeckN] `beatsync_phase`

[SamplerN] `beatsync_phase`

Syncs the phase to that of the other track (if BPM is detected on both).

Range: binary

Feedback: The Sync button flashes and the tempo slider snap to the appropriate value.

New in version 1.10.0.

[ChannelN] `beatsync_tempo`

[PreviewDeckN] `beatsync_tempo`

[SamplerN] `beatsync_tempo`

Syncs the tempo to that of the other track (if BPM is detected on both).

Range: binary

Feedback: The Sync button flashes and the tempo slider snaps to the appropriate value.

New in version 1.10.0.

[ChannelN] `bpm`

[PreviewDeckN] `bpm`

[SamplerN] `bpm`

Reflects the perceived (rate-adjusted) BPM of the loaded file.

This is a ControlPotMeter control.

Range: real-valued

Feedback: BPM value display

Changed in version 1.10.0.

[ChannelN] `bpm_tap`

[PreviewDeckN] `bpm_tap`

[SamplerN] `bpm_tap`

When tapped repeatedly, adjusts the BPM of the track on the deck (not the tempo slider!) to match the taps.

> **Note**
>
> If you want to change the rate of the deck use `[ChannelN],tempo_tap` instead.

Range:	binary
Feedback:	BPM value display (playback speed doesn’t change)

New in version 1.9.2.

[ChannelN] `tempo_tap`

[PreviewDeckN] `tempo_tap`

[SamplerN] `tempo_tap`

When tapped repeatedly, adjusts the rate/tempo of the deck to match the taps.

Range:	binary
Feedback:	Speed slider

New in version 2.5.0.

[ChannelN] `CloneFromDeck`

[PreviewDeckN] `CloneFromDeck`

[SamplerN] `CloneFromDeck`

Clone the given deck number, copying the play state, position, rate, and key. If 0 or a negative number is given, Mixxx will attempt to select the first playing deck as the source for the clone.

Range:	integer between 1 and `[Master],num_decks` (inclusive)
Feedback:	The channel will start playing at the rate and position of the source deck.

New in version 2.3.0.

[ChannelN] `CloneFromSampler`

[PreviewDeckN] `CloneFromSampler`

[SamplerN] `CloneFromSampler`

Clone the given sampler number, copying the play state, position, rate, and key.

Range:	integer between 1 and `[App],num_samplers` (inclusive)
Feedback:	The channel will start playing at the rate and position of the source deck.

New in version 2.3.0.

[ChannelN] **`LoadTrackFromDeck`**

[PreviewDeckN] **`LoadTrackFromDeck`**

[SamplerN] **`LoadTrackFromDeck`**

Load the track currently loaded to the given deck number.

Range:	integer between 1 and `[App],num_decks` (inclusive)
Feedback:	Track name & waveform change

New in version 2.4.0.

[ChannelN] **`LoadTrackFromSampler`**

[PreviewDeckN] **`LoadTrackFromSampler`**

[SamplerN] **`LoadTrackFromSampler`**

Load the track currently loaded to the given sampler number.

Range:	integer between 1 and `[App],num_samplers` (inclusive)
Feedback:	Track name & waveform change

New in version 2.4.0.

[ChannelN] **`cue_cdj`**

[PreviewDeckN] **`cue_cdj`**

[SamplerN] **`cue_cdj`**

Represents a Cue button that is always in CDJ mode.

Range:	binary
Feedback:	None

New in version 1.10.0.

[ChannelN] **`cue_clear`**

[PreviewDeckN] **`cue_clear`**

[SamplerN] **`cue_clear`**

Deletes the already set cue point and sets `[ChannelN],cue_point` to -1.

Range:	binary
Feedback:	None

[ChannelN] **`cue_goto`**

[PreviewDeckN] **`cue_goto`**

[SamplerN] **`cue_goto`**

If the cue point is set, recalls the cue point.

Range:	binary
Feedback:	Player may change position

[ChannelN] **`cue_default`**

[PreviewDeckN] **`cue_default`**

[SamplerN] **`cue_default`**

In CDJ mode, when playing, returns to the cue point and pauses. If stopped, sets a cue point at the current location. If stopped and at a cue point, plays from that point until released (set to 0.)

Range:	binary
Feedback:	Cue button

[ChannelN] **`cue_gotoandplay`**

[PreviewDeckN] **`cue_gotoandplay`**

[SamplerN] **`cue_gotoandplay`**

If the cue point is set, seeks the player to it and starts playback.

Range:	binary
Feedback:	Player may change position and start playing.

New in version 1.11.0.

[ChannelN] **`cue_gotoandstop`**

[PreviewDeckN] **`cue_gotoandstop`**

[SamplerN] **`cue_gotoandstop`**

If the cue point is set, seeks the player to it and stops.

Range:	binary
Feedback:	Player may change position.

New in version 1.11.0.

[ChannelN] **cue_indicator**

[PreviewDeckN] **cue_indicator**

[SamplerN] **cue_indicator**

Indicates the blinking pattern of the CUE button (i.e. 1.0 if the button is illuminated, 0.0 otherwise), depending on the chosen `cue mode`.

Range: binary
Feedback: Cue button

New in version 2.0.0.

[ChannelN] **cue_mode**

[PreviewDeckN] **cue_mode**

[SamplerN] **cue_mode**

Represents the currently chosen cue mode.

Range:

Value	compatible hardware
0.0	Mixxx mode (default)
1.0	Pioneer mode
2.0	Denon mode
3.0	Numark mode
4.0	Mixxx mode (no blinking)
5.0	CUP (Cue + Play) mode

Feedback: None

[ChannelN] **cue_play**

[PreviewDeckN] **cue_play**

[SamplerN] **cue_play**

Go to cue point and play after release (CUP button behavior). If stopped, sets a cue point at the current location.

Range: binary
Feedback: None

New in version 2.1.0.

[ChannelN] **`cue_point`**

[PreviewDeckN] **`cue_point`**

[SamplerN] **`cue_point`**

The current position of the cue point in samples.

Range: absolute value
Feedback: Cue point marker

[ChannelN] **`cue_preview`**

[PreviewDeckN] **`cue_preview`**

[SamplerN] **`cue_preview`**

Plays from the current cue point.

Range: binary
Feedback: Cue button lights and waveform moves

[ChannelN] **`cue_set`**

[PreviewDeckN] **`cue_set`**

[SamplerN] **`cue_set`**

Sets a cue point at the current location.

Range: binary
Feedback: Cue mark appears on the waveform

[ChannelN] **`cue_simple`**

[PreviewDeckN] **`cue_simple`**

[SamplerN] **`cue_simple`**

If the player is not playing, set the cue point at the current location otherwise seek to the cue point.

Range: binary
Feedback: Cue button

[ChannelN] **`duration`**

[PreviewDeckN] **`duration`**

[SamplerN] **`duration`**

Outputs the length of the current song in seconds

Range:	absolute value
Feedback:	None

[ChannelN] **`eject`**

[PreviewDeckN] **`eject`**

[SamplerN] **`eject`**

Eject currently loaded track. If no track is loaded the last-ejected track (of any deck) is reloaded.

Double-press to reload the last replaced track. If no track is loaded the second-last ejected track is reloaded.

Range:	binary
Feedback:	Eject button is lit. Be sure to set back to 0 with scripts so the button does not stay lit.

New in version 1.9.0.

[ChannelN] **`end`**

[PreviewDeckN] **`end`**

[SamplerN] **`end`**

Jump to end of track

Range:	binary
Feedback:	Track jumps to end

[ChannelN] **`end_of_track`**

[PreviewDeckN] **`end_of_track`**

[SamplerN] **`end_of_track`**

Switches to 1 if the play position is within the end range defined in Preferences ‣ Waveforms ‣ End of track warning.

Range:	binary, read-only
Feedback:	Waveform and Overview widgets show a flashing border

[ChannelN] **`file_bpm`**

[PreviewDeckN] **`file_bpm`**

[SamplerN] **`file_bpm`**

The detected BPM of the loaded track.

Range:	positive value, read-only
Feedback:	None

[ChannelN] **`file_key`**

[PreviewDeckN] **`file_key`**

[SamplerN] **`file_key`**

The detected key of the loaded track.

Range:	?, read-only
Feedback:	None

New in version 2.0.0.

[ChannelN] **`fwd`**

[PreviewDeckN] **`fwd`**

[SamplerN] **`fwd`**

Fast forward (FF)

Range:	binary
Feedback:	> button

[ChannelN] **`hotcue_X_activate`**

[PreviewDeckN] **`hotcue_X_activate`**

[SamplerN] **`hotcue_X_activate`**

If hotcue X is not set, this sets a hotcue at the current play position and saves it as hotcue X of `type` "Hotcue". In case a loop is currently enabled (i.e. if *[ChannelN],loop_enabled* is set to 1), the loop will be saved as hotcue X instead and `hotcue_X_type` will be set to "Loop".

If hotcue X has been set as a regular cue point, the player seeks to the saved play position.

If `hotcue_X_type` is "Loop", looping will be enabled and the loop controls (e.g. `loop_start_position`, `loop_end_position` and `beatloop_size`) will be set accordingly. Just like `reloop_toggle`, the player seeks back to the loop start when the current play position is behind the loop, and enabled without a seek when it is in front of or inside the loop. This allows a loop catching behavior on one hand and a jump back when the loop has been exit by just triggering this control.

Setting the control to 1 when the track is currently not playing (i.e. `play` is set to 0) will start hotcue previewing. After resetting the control to 0, playback will usually be stopped and the player will seek to the hotcue position. If `play` is set to 1 while previewing is active, the playback will continue and no seek occurs.

Range:	binary
Feedback:	Player may change position. Hotcue X marker may change on waveform.

New in version 1.8.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] `hotcue_X_activatecue`

[PreviewDeckN] `hotcue_X_activatecue`

[SamplerN] `hotcue_X_activatecue`

Identical to `hotcue_X_activate`, but this always sets a regular cue point, regardless of whether a loop is enabled or not. This control can be used for controllers that have dedicated hotcue/saved loop pad modes.

New in version 2.4.0.

[ChannelN] `hotcue_X_activateloop`

[PreviewDeckN] `hotcue_X_activateloop`

[SamplerN] `hotcue_X_activateloop`

Identical to `hotcue_X_activate`, but this always sets a saved loop, regardless of whether a loop is enabled or not. If no loop is available, this sets and enables a beat loop of of `beatloop_size`. This control can be used for controllers that have dedicated hotcue/saved loop pad modes.

New in version 2.4.0.

[ChannelN] `hotcue_X_cueloop`

[PreviewDeckN] `hotcue_X_cueloop`

[SamplerN] `hotcue_X_cueloop`

Enables or disables a loop from the position of hotcue X. If X is a saved loop, that loop will be used, otherwise it will set a beatloop of `beatloop_size` from the cue position. In case the hotcue is not set, this control will set a regular cue point at the current position and start a beatloop.

This control can be used to map the primary action of the “Cue Loop” performance pad mode on Serato-style controllers.

New in version 2.4.0.

[ChannelN] **`hotcue_X_clear`**

[PreviewDeckN] **`hotcue_X_clear`**

[SamplerN] **`hotcue_X_clear`**

If hotcue X is set, clears its hotcue status.

Range:	binary
Feedback:	Hotcue X marker changes on waveform.

New in version 1.8.0.

[ChannelN] **`hotcue_X_color`**

[PreviewDeckN] **`hotcue_X_color`**

[SamplerN] **`hotcue_X_color`**

Color of hotcue X or -1 if the hotcue is not set.

Range:	3-Byte RGB color code (or -1)
Feedback:	Color of Hotcue X button and waveform marker changes.

New in version 2.3.0.

[ChannelN] **`hotcue_X_status`**

[PreviewDeckN] **`hotcue_X_status`**

[SamplerN] **`hotcue_X_status`**

Indicates if hotcue slot X is set, active or empty.

Range (read-only):

Value	Meaning
0	Hotcue X is not set
1	Hotcue X is set
2	Hotcue X is active (saved loop is enabled or hotcue is previewing)

New in version 2.4.0: Replaces the deprecated `[ChannelN],hotcue_X_enabled`.

[ChannelN] **`hotcue_X_type`**

[PreviewDeckN] **`hotcue_X_type`**

[SamplerN] **`hotcue_X_type`**

Indicates the type of the hotcue in hotcue slot X.

Range (read-only):

Value	Type
0	Invalid/Not Set
1	Hotcue
4	Saved Loop

New in version 2.4.0.

[ChannelN] **hotcue_X_goto**

[PreviewDeckN] **hotcue_X_goto**

[SamplerN] **hotcue_X_goto**

If hotcue X is set, seeks the player to hotcue X's position.

Range:	binary
Feedback:	Player may change position.

New in version 1.8.0.

[ChannelN] **hotcue_X_gotoandplay**

[PreviewDeckN] **hotcue_X_gotoandplay**

[SamplerN] **hotcue_X_gotoandplay**

If hotcue X is set, seeks the player to hotcue X's position and starts playback.

Range:	binary
Feedback:	Player may change position.

New in version 1.11.0.

[ChannelN] **hotcue_X_gotoandloop**

[PreviewDeckN] **hotcue_X_gotoandloop**

[SamplerN] **hotcue_X_gotoandloop**

If hotcue X is set, seeks the player to hotcue X's position, starts playback and looping. If the hotcue is a saved loop, the loop is enabled, otherwise a beatloop of `beatloop_size` is set from the hotcue's position.

This control can be used to map the secondary action of the "Cue Loop" performance pad mode on Serato-style controllers.

Range:	binary
Feedback:	Player may change position and looping is enabled.

New in version 2.4.0.

[ChannelN] **`hotcue_X_gotoandstop`**

[PreviewDeckN] **`hotcue_X_gotoandstop`**

[SamplerN] **`hotcue_X_gotoandstop`**

If hotcue X is set, seeks the player to hotcue X’s position and stops.

Range: binary
Feedback: Player may change position.

New in version 1.8.0.

[ChannelN] **`hotcue_X_position`**

[PreviewDeckN] **`hotcue_X_position`**

[SamplerN] **`hotcue_X_position`**

The position of hotcue X in samples, -1 if not set.

Range: positive integer
Feedback: Hotcue X marker changes on waveform.

New in version 1.8.0.

[ChannelN] **`hotcue_X_set`**

[PreviewDeckN] **`hotcue_X_set`**

[SamplerN] **`hotcue_X_set`**

Set a hotcue at the current play position and saves it as hotcue X of `type` “Hotcue”. In case a loop is currently enabled (i.e. if *[ChannelN],loop_enabled* is set to 1), the loop will be saved as hotcue X instead and `hotcue_X_type` will be set to “Loop”.

Range: binary
Feedback: Hotcue X marker changes on waveform.

New in version 1.8.0.

Changed in version 2.4.0.

[ChannelN] **`hotcue_X_setcue`**

[PreviewDeckN] **`hotcue_X_setcue`**

[SamplerN] **`hotcue_X_setcue`**

Identical to `hotcue_X_set`, but this always sets a regular cue point (i.e. `hotcue_X_type` "Hotcue"), regardless of whether a loop is enabled or not.

This control can be used for controllers that have dedicated hotcue/saved loop pad modes.

New in version 2.4.0.

[ChannelN] **`hotcue_X_setloop`**

[PreviewDeckN] **`hotcue_X_setloop`**

[SamplerN] **`hotcue_X_setloop`**

Identical to `hotcue_X_set`, but this always saves a loop (i.e. `hotcue_X_type` "Loop"), regardless of whether a loop is enabled or not. If no loop is available, this sets and enables a beat loop of of `beatloop_size`. This control can be used for controllers that have dedicated hotcue/saved loop pad modes.

New in version 2.4.0.

[ChannelN] **`hotcue_focus`**

[PreviewDeckN] **`hotcue_focus`**

[SamplerN] **`hotcue_focus`**

Contains the number of the most recently used hotcue (or -1 if no hotcue was used).

Range:	positive integer (or -1)
Feedback:	None

New in version 2.3.0.

[ChannelN] **`hotcue_focus_color_prev`**

[PreviewDeckN] **`hotcue_focus_color_prev`**

[SamplerN] **`hotcue_focus_color_prev`**

If there is a focused hotcue, sets its color to the previous color in the palette.

Range:	binary
Feedback:	Color of focused hotcue button and waveform marker changes.

New in version 2.3.0.

[ChannelN] **`hotcue_focus_color_next`**

[PreviewDeckN] **`hotcue_focus_color_next`**

[SamplerN] `hotcue_focus_color_next`

If there is a focused hotcue, sets its color to the next color in the palette.

Range:	binary
Feedback:	Color of focused hotcue button and waveform marker changes.

New in version 2.3.0.

[ChannelN] `intro_end_activate`

[PreviewDeckN] `intro_end_activate`

[SamplerN] `intro_end_activate`

If the intro end cue is set, seeks the player to the intro end position. If the intro end is not set, sets the intro end to the current play position.

Range:	binary
Feedback:	Player may change position. Intro end marker may change on waveform.

New in version 2.3.0.

[ChannelN] `intro_end_clear`

[PreviewDeckN] `intro_end_clear`

[SamplerN] `intro_end_clear`

If the intro end cue is set, clears its status.

Range:	binary
Feedback:	Intro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] `intro_end_enabled`

[PreviewDeckN] `intro_end_enabled`

[SamplerN] `intro_end_enabled`

1 if intro end cue is set, (position is not -1), 0 otherwise.

Range:	binary, read-only
Feedback:	Intro end button lights up.

New in version 2.3.0.

[ChannelN] `intro_end_position`

[PreviewDeckN] **`intro_end_position`**

[SamplerN] **`intro_end_position`**

The position of the intro end in samples, -1 if not set.

Range: positive integer

Feedback: Intro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`intro_end_set`**

[PreviewDeckN] **`intro_end_set`**

[SamplerN] **`intro_end_set`**

Set intro end to the current play position. If intro end was previously set, it is moved to the new position.

Range: binary

Feedback: Intro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`intro_start_activate`**

[PreviewDeckN] **`intro_start_activate`**

[SamplerN] **`intro_start_activate`**

If the intro start cue is set, seeks the player to the intro start position. If the intro start is not set, sets the intro start to the current play position.

Range: binary

Feedback: Player may change position. Intro start marker may change on waveform.

New in version 2.3.0.

[ChannelN] **`intro_start_clear`**

[PreviewDeckN] **`intro_start_clear`**

[SamplerN] **`intro_start_clear`**

If the intro start cue is set, clears its status.

Range: binary

Feedback: Intro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`intro_start_enabled`**

[PreviewDeckN] **`intro_start_enabled`**

[SamplerN] **`intro_start_enabled`**

1 if intro start cue is set, (position is not -1), 0 otherwise.

Range: binary, read-only

Feedback: Intro start button lights up.

New in version 2.3.0.

[ChannelN] **`intro_start_position`**

[PreviewDeckN] **`intro_start_position`**

[SamplerN] **`intro_start_position`**

The position of the intro start in samples, -1 if not set.

Range: positive integer

Feedback: Intro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`intro_start_set`**

[PreviewDeckN] **`intro_start_set`**

[SamplerN] **`intro_start_set`**

Set intro start to the current play position. If intro start was previously set, it is moved to the new position.

Range: binary

Feedback: Intro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`key`**

[PreviewDeckN] **`key`**

[SamplerN] **`key`**

Current key of the track

Range:

Value	OpenKey	Lancelot	Traditional
1	1d	8b	C
2	8d	3b	D♭
3	3d	10b	D
4	10d	5b	E♭
5	5d	12b	E
6	12d	7b	F
7	7d	2b	F♯/G♭
8	2d	9b	G
9	9d	4b	A♭
10	4d	11b	A
11	11d	6b	B♭
12	6d	1b	B
13	10m	5a	Cm
14	5m	12a	C♯m
15	12m	7a	Dm
16	7m	2a	D♯m/E♭m
17	2m	9a	Em
18	9m	4a	Fm
19	4m	11a	F♯m
20	11m	6a	Gm
21	6m	1a	G♯m
22	1m	8a	Am
23	8m	3a	B♭m
24	3m	10a	Bm

New in version 2.0.0.

[ChannelN] `keylock`

[PreviewDeckN] `keylock`

[SamplerN] `keylock`

Enable key-lock for the specified deck (rate changes only affect tempo, not key)

Range:	binary
Feedback:	key-lock button activates

New in version 1.9.0.

[ChannelN] `LoadSelectedTrack`

[PreviewDeckN] `LoadSelectedTrack`

[SamplerN] `LoadSelectedTrack`

Loads the currently highlighted track into the deck

Range:	binary
Feedback:	Track name & waveform change

[ChannelN] `LoadSelectedTrackAndPlay`

[PreviewDeckN] `LoadSelectedTrackAndPlay`

[SamplerN] `LoadSelectedTrackAndPlay`

Loads the currently highlighted track into the deck and starts playing.

If the player is a preview deck and the selected track is already loaded, toggle play/pause.

Range:	binary
Feedback:	Track name & waveform change & Play/pause button

New in version 1.11.0.

[ChannelN] `local_bpm`

[PreviewDeckN] `local_bpm`

[SamplerN] `local_bpm`

Reflects the average bpm around the current play position of the loaded file.

Range:	positive value
Feedback:	None

[ChannelN] `loop_double`

[PreviewDeckN] `loop_double`

[SamplerN] `loop_double`

Doubles `beatloop_size`. If `beatloop_size` equals the size of the loop, the loop is resized.

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

Range:	binary
Feedback:	Beatloop size spinbox changes

New in version 1.10.0.

Changed in version 2.1.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_enabled`**

[PreviewDeckN] **`loop_enabled`**

[SamplerN] **`loop_enabled`**

Indicates whether or not a loop is enabled.

Range:	binary
Feedback:	Loop in waveform is active.

New in version 1.8.0.

Changed in version 2.4.0: Control is not longer read-only and can be used to enable/disable looping.

[ChannelN] **`loop_remove`**

[PreviewDeckN] **`loop_remove`**

[SamplerN] **`loop_remove`**

Clears the last active loop, i.e. deactivates and removes loop, detaches `loop_in`, `loop_out`, `reloop_toggle` and related controls. It does not affect saved loops.

Range:	binary
Feedback:	Last active loop is disabled and removed from waveform and overview.

New in version 2.4.0.

[ChannelN] **`loop_end_position`**

[PreviewDeckN] **`loop_end_position`**

[SamplerN] **`loop_end_position`**

The player loop-out position in samples, -1 if not set.

Range:	positive integer
Feedback:	Loop-out marker shows on waveform.

New in version 1.8.0.

[ChannelN] **`loop_halve`**

[PreviewDeckN] **`loop_halve`**

[SamplerN] **`loop_halve`**

Halves `beatloop_size`. If `beatloop_size` equals the size of the loop, the loop is resized.

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

Range:	binary
Feedback:	Beatloop size spinbox changes

New in version 1.10.0.

Changed in version 2.1.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_in`**

[PreviewDeckN] **`loop_in`**

[SamplerN] **`loop_in`**

If loop is disabled, sets the player loop in position to the current play position. If loop is enabled, press and hold to move loop in position to the current play position. If quantize is enabled, `beatloop_size` will be updated to reflect the new loop size.

Range:	binary
Feedback:	Loop-in marker changes on waveform.

New in version 1.8.0.

Changed in version 2.1.0.

[ChannelN] **`loop_in_goto`**

[PreviewDeckN] **`loop_in_goto`**

[SamplerN] **`loop_in_goto`**

Seek to the loop in point.

Range:	binary
Feedback:	Waveform position jumps

New in version 2.1.0.

[ChannelN] **`loop_out`**

[PreviewDeckN] **`loop_out`**

[SamplerN] **`loop_out`**

If loop is disabled, sets the player loop out position to the current play position. If loop is enabled, press and hold to move loop out position to the current play position. If quantize is enabled, `beatloop_size` will be updated to reflect the new loop size.

Range:	binary
Feedback:	Loop-out marker changes on waveform.

New in version 1.8.0.

Changed in version 2.1.0.

[ChannelN] **`loop_out_goto`**

[PreviewDeckN] **`loop_out_goto`**

[SamplerN] **`loop_out_goto`**

Seek to the loop out point.

Range:	binary
Feedback:	Waveform position jumps

New in version 2.1.0.

[ChannelN] **`loop_move`**

[PreviewDeckN] **`loop_move`**

[SamplerN] **`loop_move`**

Move loop forward by X beats (positive) or backward by X beats (negative).

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

Range:	real number
Feedback:	Loop moves forward or backward by X beats.

New in version 2.0.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_move_X_forward`**

[PreviewDeckN] **`loop_move_X_forward`**

[SamplerN] **`loop_move_X_forward`**

Moves the loop in and out points forward by X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512

Range:	binary
Feedback:	Loop moves forward by X beats.

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

New in version 2.0.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_move_X_backward`**

[PreviewDeckN] **`loop_move_X_backward`**

[SamplerN] **`loop_move_X_backward`**

Loop moves by X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

Range:	binary
Feedback:	Loop moves backward by X beats.

New in version 2.0.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_scale`**

[PreviewDeckN] **`loop_scale`**

[SamplerN] **`loop_scale`**

Scale the loop length by the value scale is set to by moving the end marker. `beatloop_size` is not updated to reflect the change.

If a saved loop is currently enabled, the modification is saved to the hotcue slot immediately.

Range:	0.0 - infinity
Feedback:	Loop length is scaled by given amount on waveform.

New in version 1.10.0.

Changed in version 2.4.0: Added support for saved loops.

[ChannelN] **`loop_start_position`**

[PreviewDeckN] **`loop_start_position`**

[SamplerN] **`loop_start_position`**

The player loop-in position in samples, -1 if not set.

Range: positive integer

Feedback: Loop-in marker changes on waveform.

New in version 1.8.0.

[ChannelN] **`orientation`**

[PreviewDeckN] **`orientation`**

[SamplerN] **`orientation`**

Set channel orientation for the crossfader.

Range:

Value	Meaning
0	Left side of crossfader
1	Center (not affected by crossfader)
2	Right side of crossfader

Feedback: None

New in version 1.9.0.

[ChannelN] **`orientation_center`**

[PreviewDeckN] **`orientation_center`**

[SamplerN] **`orientation_center`**

[AuxiliaryN] **`orientation_center`**

Assign channel to the center of the crossfader.

[ChannelN] **`orientation_left`**

[PreviewDeckN] **`orientation_left`**

[SamplerN] **`orientation_left`**

[AuxiliaryN] **`orientation_left`**

Assign channel to the left side of the crossfader.

[ChannelN] **orientation_right**

[PreviewDeckN] **orientation_right**

[SamplerN] **orientation_right**

[AuxiliaryN] **orientation_right**

Assign channel to the right side of the crossfader.

[ChannelN] **outro_end_activate**

[PreviewDeckN] **outro_end_activate**

[SamplerN] **outro_end_activate**

If the outro end cue is set, seeks the player to the outro end position. If the outro end is not set, sets the outro end to the current play position.

Range:	binary
Feedback:	Player may change position. Outro end marker may change on waveform.

New in version 2.3.0.

[ChannelN] **outro_end_clear**

[PreviewDeckN] **outro_end_clear**

[SamplerN] **outro_end_clear**

If the outro end cue is set, clears its status.

Range:	binary
Feedback:	Outro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] **outro_end_enabled**

[PreviewDeckN] **outro_end_enabled**

[SamplerN] **outro_end_enabled**

1 if outro end cue is set, (position is not -1), 0 otherwise.

Range:	binary, read-only
Feedback:	Outro end button lights up.

New in version 2.3.0.

[ChannelN] **outro_end_position**

[PreviewDeckN] `outro_end_position`

[SamplerN] `outro_end_position`

The position of the outro end in samples, -1 if not set.

Range:	positive integer
Feedback:	Outro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] `outro_end_set`

[PreviewDeckN] `outro_end_set`

[SamplerN] `outro_end_set`

Set outro end to the current play position. If outro end was previously set, it is moved to the new position.

Range:	binary
Feedback:	Outro end marker changes on waveform.

New in version 2.3.0.

[ChannelN] `outro_start_activate`

[PreviewDeckN] `outro_start_activate`

[SamplerN] `outro_start_activate`

If the outro start cue is set, seeks the player to the outro start position. If the outro start is not set, sets the outro start to the current play position.

Range:	binary
Feedback:	Player may change position. Outro start marker may change on waveform.

New in version 2.3.0.

[ChannelN] `outro_start_clear`

[PreviewDeckN] `outro_start_clear`

[SamplerN] `outro_start_clear`

If the outro start cue is set, clears its status.

Range:	binary
Feedback:	Outro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`outro_start_enabled`**

[PreviewDeckN] **`outro_start_enabled`**

[SamplerN] **`outro_start_enabled`**

1 if outro start cue is set, (position is not -1), 0 otherwise.

Range:	binary, read-only
Feedback:	Outro start button lights up.

New in version 2.3.0.

[ChannelN] **`outro_start_position`**

[PreviewDeckN] **`outro_start_position`**

[SamplerN] **`outro_start_position`**

The position of the outro start in samples, -1 if not set.

Range:	positive integer
Feedback:	Outro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`outro_start_set`**

[PreviewDeckN] **`outro_start_set`**

[SamplerN] **`outro_start_set`**

Set outro start to the current play position. If outro start was previously set, it is moved to the new position.

Range:	binary
Feedback:	Outro start marker changes on waveform.

New in version 2.3.0.

[ChannelN] **`passthrough`**

[PreviewDeckN] **`passthrough`**

[SamplerN] **`passthrough`**

Connects the vinyl control input for vinyl control on that deck to the channel output. Allows to mix external media into DJ sets.

Range:	binary
Feedback:	Passthrough label in the waveform overview and passthrough button

New in version 2.0.0.

[ChannelN] **`PeakIndicator`**

[PreviewDeckN] **`PeakIndicator`**

[SamplerN] **`PeakIndicator`**

Indicates when the signal is clipping (too loud for the hardware and is being distorted)

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light

[ChannelN] **`PeakIndicatorL`**

[PreviewDeckN] **`PeakIndicatorL`**

[SamplerN] **`PeakIndicatorL`**

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the left channel

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (left)

New in version 2.0.0.

[ChannelN] **`PeakIndicatorR`**

[PreviewDeckN] **`PeakIndicatorR`**

[SamplerN] **`PeakIndicatorR`**

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the right channel

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (right)

New in version 2.0.0.

[ChannelN] **`pfl`**

[PreviewDeckN] **`pfl`**

[SamplerN] **`pfl`**

Toggles headphone cueing (PFL).

Range:	binary
Feedback:	Headphone button

[ChannelN] **`pitch`**

[PreviewDeckN] **`pitch`**

[SamplerN] **`pitch`**

The total adjustment to the track's pitch, including changes from the rate slider if keylock is off as well as `pitch_adjust`.

> **Note**
>
> Do not map this to knobs or sliders on controllers; map `pitch_adjust` instead.

This is a ControlPotMeter control.

Range:	-6.0..6.0 semitones
Feedback:	Key display

New in version 2.0.0.

[ChannelN] **`pitch_up`**

[PreviewDeckN] **`pitch_up`**

[SamplerN] **`pitch_up`**

Changes the track pitch up one half step, independent of the tempo.

Range:	binary
Feedback:	Key display

New in version 2.0.0.

[ChannelN] **`pitch_down`**

[PreviewDeckN] **`pitch_down`**

[SamplerN] **`pitch_down`**

Changes the track pitch down one half step, independent of the tempo.

Range:	binary
Feedback:	Key display

New in version 2.0.0.

[ChannelN] **`pitch_adjust`**

[PreviewDeckN] **`pitch_adjust`**

[SamplerN] **`pitch_adjust`**

Adjusts the pitch in addition to the tempo slider pitch and keylock. It is reset after loading a new track.

This is a ControlPotMeter control.

Range:	-3.0..3.0 semitones
Feedback:	Key display

New in version 2.0.0.

[ChannelN] **`play`**

[PreviewDeckN] **`play`**

[SamplerN] **`play`**

Toggles playing or pausing the track.

The value is set to 1 when the track is playing or when previewing from cue points and when the play command is adopted and track will be played after loading.

Range:	binary
Feedback:	Play/pause button

[ChannelN] **`play_indicator`**

[PreviewDeckN] **`play_indicator`**

[SamplerN] **`play_indicator`**

Provides information to be bound with the a Play/Pause button e.g blinking when play is possible

Range:	binary, read-only
Feedback:	Play/pause button

New in version 2.0.0.

[ChannelN] **`play_latched`**

[PreviewDeckN] **`play_latched`**

[SamplerN] **`play_latched`**

This is set to 1 when the track is playing, but not when previewing (see `play`).

Range:	binary, read-only
Feedback:	Play/pause button

New in version 2.3.0.

[ChannelN] `play_stutter`

[PreviewDeckN] `play_stutter`

[SamplerN] `play_stutter`

A play button without pause. Pushing while playing, starts play at cue point again (Stutter).

Range:	binary
Feedback:	Play/Stutter button

New in version 2.0.0.

[ChannelN] `playposition`

[PreviewDeckN] `playposition`

[SamplerN] `playposition`

Sets the absolute position in the track.

This is a ControlPotMeter control.

Range:	-0.14 to 1.14 (0 = beginning -> Midi 14, 1 = end -> Midi 114)
Feedback:	Waveform

[ChannelN] `pregain`

[PreviewDeckN] `pregain`

[SamplerN] `pregain`

Adjusts the pre-fader gain of the track (to avoid clipping)

This is a ControlPotMeter control.

Range:	0.0..1.0..4.0
Feedback:	GAIN knob

[ChannelN] `quantize`

[PreviewDeckN] `quantize`

[SamplerN] `quantize`

Aligns Hot-cues and Loop In & Out to the next beat from the current position.

Range:	binary

Feedback:	Hot-cues or Loop In/Out markers

New in version 1.10.0.

[ChannelN] `rate`

[PreviewDeckN] `rate`

[SamplerN] `rate`

Speed control

This is a ControlPotMeter control.

Range:	-1.0..1.0
Feedback:	Speed slider

[ChannelN] `rate_dir`

[PreviewDeckN] `rate_dir`

[SamplerN] `rate_dir`

Indicates orientation of speed slider.

Range:	-1 or 1

[ChannelN] `rate_perm_down`

[PreviewDeckN] `rate_perm_down`

[SamplerN] `rate_perm_down`

Sets the speed one step lower (4 % default) lower

Range:	binary
Feedback:	Perm down button & Speed slider

[ChannelN] `rate_perm_down_small`

[PreviewDeckN] `rate_perm_down_small`

[SamplerN] `rate_perm_down_small`

Sets the speed one small step lower (1 % default)

Range:	binary
Feedback:	Perm down button & Speed slider

[ChannelN] `rate_perm_up`

[PreviewDeckN] `rate_perm_up`

[SamplerN] `rate_perm_up`

Sets the speed one step higher (4 % default)

Range:	binary
Feedback:	Perm up button & Speed slider

[ChannelN] `rate_perm_up_small`

[PreviewDeckN] `rate_perm_up_small`

[SamplerN] `rate_perm_up_small`

Sets the speed one small step higher (1 % default)

Range:	binary
Feedback:	Perm up button & Speed slider

[ChannelN] `rate_temp_down`

[PreviewDeckN] `rate_temp_down`

[SamplerN] `rate_temp_down`

Holds the speed one step lower while active

Range:	binary
Feedback:	Temp down button & Speed slider

[ChannelN] `rate_temp_down_small`

[PreviewDeckN] `rate_temp_down_small`

[SamplerN] `rate_temp_down_small`

Holds the speed one small step lower while active

Range:	binary
Feedback:	Temp down button & Speed slider

[ChannelN] `rate_temp_up`

[PreviewDeckN] `rate_temp_up`

[SamplerN] `rate_temp_up`

Holds the speed one step higher while active

Range:	binary
Feedback:	Temp up button & Speed slider

[ChannelN] `rate_temp_up_small`

[PreviewDeckN] `rate_temp_up_small`

[SamplerN] `rate_temp_up_small`

Holds the speed one small step higher while active

Range:	binary
Feedback:	Temp up button & Speed slider

[ChannelN] `rateRange`

[PreviewDeckN] `rateRange`

[SamplerN] `rateRange`

Sets the range of the Speed slider (0.08 = 8%)

This is a ControlPotMeter control.

Range:	0.0..4.0
Feedback:	none, until you move the Speed slider

[ChannelN] `rateSearch`

[PreviewDeckN] `rateSearch`

[SamplerN] `rateSearch`

Seeks forward (positive values) or backward (negative values) at a speed determined by the value

This is a ControlPotMeter control.

Range:	-300..300
Feedback:	Deck seeks

[ChannelN] `rateEngine`

[PreviewDeckN] `rateEngine`

[SamplerN] `rateEngine`

Actual rate (used in visuals, not for control)

[ChannelN] `reloop_andstop`

[PreviewDeckN] `reloop_andstop`

[SamplerN] `reloop_andstop`

Activate current loop, jump to its loop in point, and stop playback.

Range:	binary

Feedback: Loop range in waveform activates or deactivates and play position moves to loop in point.

New in version 2.1.0.

[ChannelN] `reloop_toggle`

[PreviewDeckN] `reloop_toggle`

[SamplerN] `reloop_toggle`

Toggles the current loop on or off. If the loop is ahead of the current play position, the track will keep playing normally until it reaches the loop.

Range: binary
Feedback: Loop range in waveform activates or deactivates.

New in version 2.1.0.

[ChannelN] `repeat`

[PreviewDeckN] `repeat`

[SamplerN] `repeat`

Enable repeat-mode for the specified deck

Range: binary
Feedback: when track finishes, song loops to beginning

New in version 1.9.0.

[ChannelN] `reset_key`

[PreviewDeckN] `reset_key`

[SamplerN] `reset_key`

Resets the key to the original track key.

Range: binary

New in version 2.0.0.

[ChannelN] `reverse`

[PreviewDeckN] `reverse`

[SamplerN] `reverse`

Toggles playing the track backwards

Range:	binary
Feedback:	REV button

[ChannelN] **`reverseroll`**

[PreviewDeckN] **`reverseroll`**

[SamplerN] **`reverseroll`**

Enables reverse and slip mode while held (Censor)

Range:	binary
Feedback:	REV button

New in version 2.0.0.

[ChannelN] **`scratch2`**

[PreviewDeckN] **`scratch2`**

[SamplerN] **`scratch2`**

Affects absolute play speed & direction whether currently playing or not when `[ChannelN],scratch2_enable` is active. (multiplicative). Use JavaScript `engine.scratch` functions to manipulate in controller mappings.

Range:	-3.0..3.0
Feedback:	Waveform

New in version 1.8.0.

[ChannelN] **`scratch2_enable`**

[PreviewDeckN] **`scratch2_enable`**

[SamplerN] **`scratch2_enable`**

Takes over play speed & direction for `[ChannelN],scratch2`.

Range:	binary
Feedback:	Waveform

New in version 1.8.0.

[ChannelN] **`show_track_menu`**

Toggle the track context menu for the track currently loaded in this deck. The control value is *1* if there is already a menu shown for this deck. The menu can be navigated with the `MoveUp/Down` controls and selected actions or submenus can be activated with `GoToItem`.

Range: Binary
Feedback: The deck's track context menu is shown or hidden.

New in version 2.5.0.

[ChannelN] `slip_enabled`

[PreviewDeckN] `slip_enabled`

[SamplerN] `slip_enabled`

Toggles slip mode. When active, the playback continues muted in the background during a loop, scratch etc. Once disabled, the audible playback will resume where the track would have been.

Range: binary
Feedback: Slip mode button

New in version 1.11.0.

[ChannelN] `stars_up`

[PreviewDeckN] `stars_up`

[SamplerN] `stars_up`

Increase the rating of the currently loaded track (if the skin has star widgets in the decks section).

Range: binary
Feedback: Star count is increased in the deck's star widget and in the library table.

New in version 2.3.0.

[ChannelN] `stars_down`

[PreviewDeckN] `stars_down`

[SamplerN] `stars_down`

Decrease the rating of the currently loaded track (if the skin has star widgets in the decks section).

Range: binary
Feedback: Star count is decreased in the deck's star widget and in the library table.

New in version 2.3.0.

[ChannelN] `start`

[PreviewDeckN] **start**

[SamplerN] **start**

Jump to start of track

Range:	binary
Feedback:	Track jumps to start

[ChannelN] **start_play**

[PreviewDeckN] **start_play**

[SamplerN] **start_play**

Start playback from the beginning of the deck.

Range:	binary
Feedback:	Deck plays from beginning

New in version 1.10.0.

[ChannelN] **start_stop**

[PreviewDeckN] **start_stop**

[SamplerN] **start_stop**

Seeks a player to the start and then stops it.

Range:	binary
Feedback:	Deck stops at the beginning.

New in version 1.10.0.

[ChannelN] **stop**

[PreviewDeckN] **stop**

[SamplerN] **stop**

Stops a player.

Range:	binary
Feedback:	Pause Button. Deck pauses at the current position.

New in version 1.10.0.

[ChannelN] **sync_enabled**

[PreviewDeckN] **sync_enabled**

[SamplerN] `sync_enabled`

Syncs the tempo and phase (depending on quantize) to that of the other track (if BPM is detected on both). Click and hold for at least one second activates sync lock on that deck.

Range: binary

Feedback: If enabled, the Sync button stays lit and tempo slider snap to the appropriate value. Slider adjustments are linked on all decks that have sync lock enabled.

New in version 2.0.0.

[ChannelN] `sync_leader`

[PreviewDeckN] `sync_leader`

[SamplerN] `sync_leader`

Sets deck as leader clock.

Range: binary

Feedback: If enabled, the Sync button stays lit and tempo slider snap to the appropriate value. Slider adjustments are linked on all decks that have sync lock enabled.

New in version 2.4.0.

[ChannelN] `sync_mode`

[PreviewDeckN] `sync_mode`

[SamplerN] `sync_mode`

New in version 2.0.0.

Range:

Value	Meaning
0	Sync lock disabled for that deck
1	Deck is sync follower
2	Deck is sync leader

[ChannelN] `sync_key`

[PreviewDeckN] `sync_key`

[SamplerN] `sync_key`

Feedback: Key value widget

Match musical key.

New in version 2.0.0.

[ChannelN] `track_color`

[PreviewDeckN] `track_color`

[SamplerN] `track_color`

Color of the currently loaded track or -1 if no track is loaded or the track has no color.

Range:	3-Byte RGB color code (or -1)
Feedback:	Track color changes in the library view.

New in version 2.3.0.

[ChannelN] `track_loaded`

[PreviewDeckN] `track_loaded`

[SamplerN] `track_loaded`

Whether a track is loaded in the specified deck

Range:	binary, read-only
Feedback:	Waveform and track metadata shown in deck

New in version 2.1.0.

[ChannelN] `track_samplerate`

[PreviewDeckN] `track_samplerate`

[SamplerN] `track_samplerate`

Sample rate of the track loaded on the specified deck

Range:	absolute value, read-only
Feedback:	None

New in version 1.9.0.

[ChannelN] `track_samples`

[PreviewDeckN] `track_samples`

[SamplerN] `track_samples`

Number of sound samples in the track loaded on the specified deck

Range:	absolute value, read-only

Feedback:	None

[ChannelN] `volume`

[PreviewDeckN] `volume`

[SamplerN] `volume`

Adjusts the channel volume fader

This is a ControlPotMeter control.

Range:	default
Feedback:	Deck volume fader

[ChannelN] `mute`

[PreviewDeckN] `mute`

[SamplerN] `mute`

Mutes the channel

Range:	binary
Feedback:	Mute button

New in version 2.0.0.

[ChannelN] `update_replaygain_from_pregain`

[PreviewDeckN] `update_replaygain_from_pregain`

[SamplerN] `update_replaygain_from_pregain`

Applies the deck pregain knob value to the detected ReplayGain value for the current track. This is a way to update the ReplayGain value of a track if it has been detected incorrectly. When this control is triggered, the pregain value for the deck will be centered so that there is no audible difference in track volume, so this operation is safe to use during performance, if the controller mapping uses soft-takeover for the pregain knob.

> **❗ Note**
>
> The pregain is adjusted only on the deck where the control was triggered. If the adjusted track is also currently playing in other decks, their effective volume will change.

Range:	binary
Feedback:	ReplayGain value is updated in library, deck pregain is reset to 1.0.

[ChannelN] `vinylcontrol_enabled`

[PreviewDeckN] **`vinylcontrol_enabled`**

[SamplerN] **`vinylcontrol_enabled`**

Toggles whether a deck is being controlled by digital vinyl.

Range: binary

Feedback: When enabled, a vinyl indication should appear onscreen indicating green for enabled.

New in version 1.10.0.

[ChannelN] **`vinylcontrol_cueing`**

[PreviewDeckN] **`vinylcontrol_cueing`**

[SamplerN] **`vinylcontrol_cueing`**

Determines how cue points are treated in vinyl control relative mode.

Range:

Value	Meaning
0	Cue points ignored
1	One Cue - If needle is dropped after the cue point, track will seek to that cue point
2	Hot Cue - Track will seek to nearest previous hotcue

New in version 1.10.0.

[ChannelN] **`vinylcontrol_mode`**

[PreviewDeckN] **`vinylcontrol_mode`**

[SamplerN] **`vinylcontrol_mode`**

Determines how vinyl control interprets needle information.

Range:

Value	Meaning
0	Absolute Mode (track position equals needle position and speed)
1	Relative Mode (track tempo equals needle speed regardless of needle position)
2	Constant Mode (track tempo equals last known-steady tempo regardless of needle input

See Control Mode for details.

Feedback: 3-way button indicates status

New in version 1.10.0.

[ChannelN] **`vinylcontrol_status`**

[PreviewDeckN] **`vinylcontrol_status`**

[SamplerN] **`vinylcontrol_status`**

Provides visual feedback with regards to vinyl control status.

Range: 0.0-3.0, read-only

Feedback: Off for control disabled, green for control enabled, blinking yellow for when the needle reaches the end of the record, and red for needle skip detected

New in version 1.10.0.

[ChannelN] **`visual_bpm`**

[PreviewDeckN] **`visual_bpm`**

[SamplerN] **`visual_bpm`**

BPM to display in the GUI (updated more slowly than the actual BPM).

Range: ?

Feedback: BPM value widget

New in version 2.0.0.

[ChannelN] **`visual_key`**

[PreviewDeckN] **`visual_key`**

[SamplerN] **`visual_key`**

Current musical key after pitch shifting to display in the GUI using the notation selected in the preferences

Range: ?

Feedback: Key value widget

New in version 2.0.0.

[ChannelN] **`visual_key_distance`**

[PreviewDeckN] **`visual_key_distance`**

[SamplerN] **`visual_key_distance`**

The distance to the nearest key measured in cents

This is a ControlPotMeter control.

Range:	-0.5..0.5
Feedback:	Key value widget

New in version 2.0.0.

[ChannelN] **`VuMeter`**

[PreviewDeckN] **`VuMeter`**

[SamplerN] **`VuMeter`**

Outputs the current instantaneous deck volume

This is a ControlPotMeter control.

Range:	default
Feedback:	Deck VU meter

[ChannelN] **`VuMeterL`**

[PreviewDeckN] **`VuMeterL`**

[SamplerN] **`VuMeterL`**

Outputs the current instantaneous deck volume for the left channel

This is a ControlPotMeter control.

Range:	default
Feedback:	Deck VU meter L

[ChannelN] **`VuMeterR`**

[PreviewDeckN] **`VuMeterR`**

[SamplerN] **`VuMeterR`**

Outputs the current instantaneous deck volume for the right channel

This is a ControlPotMeter control.

Range:	default
Feedback:	Deck VU meter R

[ChannelN] **`waveform_zoom`**

[PreviewDeckN] **`waveform_zoom`**

[SamplerN] **`waveform_zoom`**

Zooms the waveform to look ahead or back as needed.

Range:	1.0 - 10.0
Feedback:	Waveform zoom buttons

New in version 1.11.0.

[ChannelN] `waveform_zoom_up`

[PreviewDeckN] `waveform_zoom_up`

[SamplerN] `waveform_zoom_up`

Waveform Zoom Out

Range:	?
Feedback:	Waveform zoom buttons

New in version 1.11.0.

[ChannelN] `waveform_zoom_down`

[PreviewDeckN] `waveform_zoom_down`

[SamplerN] `waveform_zoom_down`

Waveform Zoom In

Range:	?
Feedback:	Waveform zoom buttons

New in version 1.11.0.

[ChannelN] `waveform_zoom_set_default`

[PreviewDeckN] `waveform_zoom_set_default`

[SamplerN] `waveform_zoom_set_default`

Return to default waveform zoom level

Range:	?
Feedback:	Waveform zoom buttons

New in version 1.11.0.

[ChannelN] `wheel`

[PreviewDeckN] `wheel`

[SamplerN] `wheel`

Affects relative playback speed and direction persistently (additive offset & must manually be undone).

Range:	-3.0..3.0
Feedback:	Waveform

Global Sampler controls

These controls can be used to control all samplers.

[Sampler] `SaveSamplerBank`

Save sampler configuration. Make currently loaded tracks in samplers instantly available at a later point.

Range:	binary
Feedback:	Opens file dialog. Configuration file can be named and saved.

New in version 2.0.0.

[Sampler] `LoadSamplerBank`

Load saved sampler configuration file and add tracks to the available samplers.

Range:	binary
Feedback:	Opens file dialog. Select configuration file.

New in version 2.0.0.

> **! See also**
>
> The `[Skin],show_samplers` control can be used to show and hide sampler banks in the GUI.

Microphones and Auxiliary Channels

In contrast to decks, preview decks and samplers, microphones and auxiliary channels are input channels. You can map audio interface’s inputs to mixxx’s auxiliary input channels and connect external audio source to it (cellphone, mp3 player). Then you can use your MIDI controller to control its volume and some other parameters (orientation, gain, volume), apply effects and use the prelisten function.

> **! Note**
>
> Although the first auxiliary group is named `[Auxiliary1]`, the group for the first microphone is just called `[Microphone]`, not `[Microphone1]`.

[ChannelN] `input_configured`

[MicrophoneN] `input_configured`

[AuxiliaryN] `input_configured`

1 if there is input is configured for this channel, 0 if not. In the case of `[ChannelN]` it corresponds to Vinyl Control. A configured input is required to enable `[ChannelN],passthrough`

Range: binary, read-only

Feedback: Configured channel in the sound preferences.

[MicrophoneN] `main_mix`

[AuxiliaryN] `main_mix`

Hold value at 1 to mix channel input into the main output. For `[MicrophoneN]` use `[MicrophoneN],talkover` instead. Note that `[AuxiliaryN]` also take `[AuxiliaryN],orientation` into account.

Range: binary

Feedback: Auxiliary: Play button Microphone: N/A

[AuxiliaryN] `orientation`

Set channel orientation for the crossfader.

Range:

Value	Meaning
0	Left side of crossfader
1	Center (not affected by crossfader)
2	Right side of crossfader

Feedback: None

New in version 1.10.0.

[MicrophoneN] `PeakIndicator`

[AuxiliaryN] `PeakIndicator`

Indicates when the signal is clipping (too loud for the hardware and is being distorted)

This is a ControlPotMeter control.

Range: binary

Feedback: Microphone Clip light

New in version 1.10.0.

[MicrophoneN] `PeakIndicatorL`

[AuxiliaryN] `PeakIndicatorL`

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the left channel

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (left)

New in version 2.0.0.

[MicrophoneN] `PeakIndicatorR`

[AuxiliaryN] `PeakIndicatorR`

Indicates when the signal is clipping (too loud for the hardware and is being distorted) for the right channel

This is a ControlPotMeter control.

Range:	binary
Feedback:	Clip light (right)

New in version 2.0.0.

[MicrophoneN] `pfl`

[AuxiliaryN] `pfl`

Toggles headphone cueing (PFL).

Range:	binary
Feedback:	Headphone button

[MicrophoneN] `talkover`

[AuxiliaryN] `talkover`

Hold value at 1 to mix channel input into the main output. For `[AuxiliaryN]` use `[AuxiliaryN],main_mix` instead. Note that `[AuxiliaryN]` also take `[AuxiliaryN],orientation` into account.

Range:	binary
Feedback:	Microphone: Talk button Auxiliary: N/A

New in version 1.10.0.

[MicrophoneN] **volume**

[AuxiliaryN] **volume**

Adjusts the channel volume fader

This is a ControlPotMeter control.

Range:	default
Feedback:	Microphone volume fader changes

New in version 1.10.0.

[MicrophoneN] **pregain**

[AuxiliaryN] **pregain**

Adjusts the gain of the input

This is a ControlPotMeter control.

Range:	0.0..1.0..4.0
Feedback:	Microphone gain knob

[MicrophoneN] **mute**

[AuxiliaryN] **mute**

Mutes the channel

Range:	binary
Feedback:	Mute button

New in version 2.0.0.

[MicrophoneN] **VuMeter**

[AuxiliaryN] **VuMeter**

Outputs the current instantaneous channel volume

This is a ControlPotMeter control.

Range:	default
Feedback:	Microphone VU meter changes

New in version 1.10.0.

[MicrophoneN] **VuMeterL**

[AuxiliaryN] **VuMeterL**

Outputs the current instantaneous deck volume for the left channel

This is a ControlPotMeter control.

Range: default
Feedback: Deck VU meter L

New in version 2.0.0.

[MicrophoneN] **`VuMeterR`**

[AuxiliaryN] **`VuMeterR`**

Outputs the current instantaneous deck volume for the right channel

This is a ControlPotMeter control.

Range: default
Feedback: Deck VU meter R

New in version 2.0.0.

The `[VinylControl]` group

The `[VinylControl]` group can toggle the vinyl control feature.

[VinylControl] **`Toggle`**

Moves control by a vinyl control signal from one deck to another if using the single deck vinyl control (VC) feature.

Range: binary
Feedback: If VC isn't enabled on any decks, enable it on the first one we're receiving samples for. If VC is enabled on a single (exclusive) deck, and another deck is setup to receive samples, disable it on the former deck and enable it on the next eligible deck (ordered by deck number). If VC is enabled on multiple decks, don't do anything.

New in version 1.10.0.

[VinylControl] **`gain`**

Allows to amplify the "phono" level of attached turntables to "line" level. This is equivalent to setting the turntable boost in Options ‣ Preferences ‣ Vinyl Control

Range: binary
Feedback: position of Boost slider in Options ‣ Preferences ‣ Vinyl Control (is not updated while viewing this Preferences page)

New in version 1.10.0.

❗ See also

The `[Skin],show_vinylcontrol` control can be used to show and hide vinyl control related elements in the GUI.

The `[Recording]` controls

The controls in the `[Recording]` group can be used to query and control the recording of your mix.

[Recording] `toggle_recording`

Turns recording on or off.

Range: binary

Feedback: Recording icon

[Recording] `status`

Indicates whether Mixxx is currently recording.

Range:

Value	Meaning
0	Recording Stopped
1	Initialize Recording
2	Recording Active

Feedback: Recording icon

AutoDJ controls

The `[AutoDJ]` controls allow interacting with AutoDJ.

[AutoDJ] `enabled`

Turns Auto DJ on or off.

Range: binary

Feedback: AutoDJ button

New in version 1.11.0.

[AutoDJ] `shuffle_playlist`

Shuffles the content of the Auto DJ playlist.

Range:	binary
Feedback:	Order of tracks in the AutoDJ playlist changes.

New in version 1.11.0.

[AutoDJ] `skip_next`

Skips the next track in the Auto DJ playlist.

Range:	binary
Feedback:	Skipped track is removed from the AutoDJ playlist.

New in version 1.11.0.

[AutoDJ] `fade_now`

Triggers the transition to the next track.

Range:	binary
Feedback:	Crossfader slider moves to the other side.

New in version 1.11.0.

[AutoDJ] `add_random_track`

Adds a random track to the Auto DJ queue.

Range:	binary
Feedback:	Track is added to AutoDJ queue.

New in version 2.4.0.

The `[Library]` controls

The controls in the `[Library]` group can be used to navigate the library. Note that `[Library],MoveUp` and other Move and Scroll controls emulate keypresses and therefore require the Mixxx window to be focused.

[Library] `MoveUp`

Equivalent to pressing the `Up` key on the keyboard

Range:	Binary
Feedback:	Currently selected item changes

New in version 2.1.0.

[Library] `MoveDown`

Equivalent to pressing the `Down` key on the keyboard

Range: Binary

Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `MoveVertical`

Move the specified number of locations up or down. Intended to be mapped to an encoder knob.

Range: Relative (positive values move down, negative values move up)

Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `ScrollUp`

Equivalent to pressing the `PageUp` key on the keyboard

Range: Binary

Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `ScrollDown`

Equivalent to pressing the `PageDown` key on the keyboard

Range: Binary

Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `ScrollVertical`

Scroll the specified number of pages up or down. Intended to be mapped to an encoder knob.

Range: Relative (positive values move down, negative values move up)

Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `MoveLeft`

Equivalent to pressing the `Left` key on the keyboard

Range: Binary
Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `MoveRight`

Equivalent to pressing the `Right` key on the keyboard

Range: Binary
Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `MoveHorizontal`

Move the specified number of locations left or right. Intended to be mapped to an encoder knob.

Range: Relative (positive values move right, negative values move left)
Feedback: Currently selected item changes

New in version 2.1.0.

[Library] `MoveFocusForward`

Equivalent to pressing the `Tab` key on the keyboard

Range: Binary
Feedback: Currently focused pane changes

New in version 2.1.0.

[Library] `MoveFocusBackward`

Equivalent to pressing the `Shift` + `Tab` key on the keyboard

Range: Binary
Feedback: Currently focused pane changes

New in version 2.1.0.

[Library] `MoveFocus`

Move focus the specified number of panes forward or backwards. Intended to be mapped to an encoder knob.

Range: Relative (positive values move forward, negative values move backward)
Feedback: Currently focused pane changes

New in version 2.1.0.

[Library] `focused_widget`

Read this control to know which library widget is currently focused, or write in order to focus a specific library widget.

This control can be used in controller scripts to trigger context-specific actions. For example, if the tracks table has focus, pressing a button loads the selected track to a specific deck, while the same button would clear the search if the search bar is focused.

Note: This control is useful only if a Mixxx window has keyboard focus, otherwise it always returns 0.

Range:

Value	writeable	Widget
0		none
1	X	Search bar
2	X	Tree view
3	X	Tracks table or root views of library features
4		Context menu (menus of library widgets or other editable widgets, or main menu bar)
5		Dialog (any confirmation or error popup, preferences, track properties or cover art window)
6		Unknown (widgets that don't fit into any of the above categories)

Feedback: Currently focused widget changes

New in version 2.4.0.

[Library] `GoToItem`

Triggers different actions, depending on which interface element currently has keyboard focus:

Search bar	
text box	moves focus to tracks table
Clear button	clears search text
Sidebar	
collapsed node	expands the item (except Tracks and Auto DJ)
leaf node	moves focus to tracks table

Tracks table	Performs the action selected in Preferences ‣ Library ‣ Track Double-Click Action (default is "Load selected track"). Also see Preferences ‣ Decks ‣ Playing track protection
Context menus	presses `Enter`
Dialogs / popups	presses `Enter`. Note: the `Move..` controls allow to move button focus.

Range: Binary
Feedback: Context dependent

New in version 2.1.0.

[Library] `show_track_menu`

Toggle the track context menu for all tracks selected in the current library view. The control value is *1* if there is already a menu shown for the current view. Note that the control is not aware of other track menus, for example those opened by right-clicking track text labels in decks. Only the most recent menu can be navigated with the `MoveUp/Down` controls and selected actions or submenus can be activated with `GoToItem`.

Range: Binary
Feedback: Tracks table context menu is shown or hidden.

New in version 2.4.0.

[Library] `AutoDjAddBottom`

[Playlist] `AutoDjAddBottom`

Add selected track(s) to Auto DJ Queue (bottom).

Range: Binary
Feedback: Append track(s) to Auto DJ playlist

New in version 2.0.0.

[Library] `AutoDjAddTop`

[Playlist] `AutoDjAddTop`

Add selected track(s) to Auto DJ Queue (top).

Range: Binary
Feedback: Prepend track(s) to Auto DJ playlist

New in version 2.0.0.

[Library] `font_size_increment`

Increase the size of the library font. If the row height is smaller than the font-size the larger of the two is used.

Range:	Binary
Feedback:	Library view

New in version 2.0.0.

[Library] `font_size_decrement`

Decrease the size of the library font

New in version 2.0.0.

Range:	Binary
Feedback:	Library view

[Library] `font_size_knob`

Increase or decrease the size of the library font

New in version 2.0.0.

Range:	Relative
Feedback:	Library view

[Library] `sort_column`

Indicates the sorting column the track table

Range:

Value	Description	Library	Playlist	Crate	Browse
1	Artist	X	X	X	X
2	Title	X	X	X	X
3	Album	X	X	X	X
4	Albumartist	X	X	X	X
5	Year	X	X	X	X
6	Genre	X	X	X	X
7	Composer	X	X	X	X
8	Grouping	X	X	X	X
9	Tracknumber	X	X	X	X
10	Filetype	X	X	X	X
11	Native Location	X	X	X	X
12	Comment	X	X	X	X
13	Duration	X	X	X	X
14	Bitrate	X	X	X	X
15	BPM	X	X	X	X
16	ReplayGain	X	X	X	X
17	Datetime Added	X	X	X	X
18	Times Played	X	X	X	X
19	Rating	X	X	X	X
20	Key	X	X	X	X
21	Preview	X	X	X	X
22	Coverart	X	X	X	
23	Position		X		
24	Playlist ID		X		
25	Location		X		
26	Filename				X
27	File Modified Time				X
28	File Creation Time				X
29	Sample Rate				
30	Track Color	X	X	X	
31	Last Played	X	X	X	

Feedback:	Sorting indicator in the column headers of the track table

New in version 2.3.0.

[Library] `sort_column_toggle`

Equivalent to clicking on column headers. A new value sets `[Library],sort_column` to that value and `[Library],sort_order` to 0, setting the same value again will toggle `[Library],sort_order`.

Range:	Same as for `[Library],sort_column` or value 0 for sorting according the current column with the cursor on it
Feedback:	Sorting indicator in the column headers of the track table

New in version 2.3.0.

[Library] `sort_order`

Indicate the sort order of the track tables.

Range:	Binary (0 for ascending, 1 for descending)
Feedback:	Sorting indicator in the column headers of the track table

New in version 2.3.0.

[Library] `sort_focused_column`

Sort the column of the table cell that is currently focused, which is equivalent to setting `[Library],sort_column_toggle` to 0. Though unlike that, it can be mapped to pushbuttons directly.

Range:	Binary
Feedback:	Sorting indicator in the column headers of the track table

New in version 2.4.0.

[Library] `track_color_prev`

Set color of selected track to previous color in palette.

Range:	Binary
Feedback:	Track color changes in the library view.

New in version 2.3.0.

[Library] `track_color_next`

Set color of selected track to next color in palette.

Range:	Binary
Feedback:	Track color changes in the library view.

New in version 2.3.0.

[Library] `search_history_next`

Select the next saved search query. Wraps around at the last item to the empty search.

Range:	Binary
Feedback:	Searchbox query changes

New in version 2.4.0.

[Library] `search_history_prev`

Select the previous saved search query. Wraps around at the top to the last item.

Range:	Binary
Feedback:	Searchbox query changes

New in version 2.4.0.

[Library] `search_history_selector`

Select another saved search query. < 0 goes up the list, > 0 goes down. Wraps around at the top and bottom.

Range:	-N / +N
Feedback:	Searchbox query changes

New in version 2.4.0.

[Library] `clear_search`

Clear the search.

Range:	Binary
Feedback:	Searchbox query is cleared

New in version 2.4.0.

See also

The library section in the GUI can be enlarged by using the `[Skin],show_maximized_library` control.

The `[Skin],show_library_coverart` control can be used to toggle the display of library coverart.

The `[Shoutcast]` controls

[Shoutcast] **`enabled`**

Shows if live Internet broadcasting is enabled.

Range:	?
Feedback:	shoutcast only supports mp3 format as field

[Shoutcast] **`status`**

This control displays whether broadcasting connection to Shoutcast server was successfully established.

Range:	binary
Feedback:	None

The `[Playlist]` controls

`[Playlist]` controls allow navigating the sidebar and tracks table directly without considering the currently focused widget. This is helpful when another application's window is focused. This group is going to be deprecated at some point, with its controls added to `[Library]` above.

❗ **See also**

See bug #1772184 for the current status.

[Playlist] **`SelectPlaylist`**

Scrolls the given number of items (view, playlist, crate, etc.) in the side pane (can be negative for reverse direction).

Range:	relative value
Feedback:	Library sidebar highlight

[Playlist] **`SelectTrackKnob`**

Scrolls the given number of tracks in the track table (can be negative for reverse direction).

Range:	relative value
Feedback:	Library track table highlight

The `[Controls]` controls

The `[Controls]` group contains controls that didn't fit in any other group.

[Controls] `touch_shift`

Once enabled, all touch tab events are interpreted as right click. This control has been added to provide touchscreen compatibility in 2.0 and might be replaced by a general modifier solution in the future.

Range:	binary
Feedback:	All Widgets

New in version 2.0.0.

[Controls] `AutoHotcueColors`

If enabled, colors will be assigned to newly created hotcues automatically.

Range:	binary
Feedback:	None

New in version 2.3.0.

[Controls] `ShowDurationRemaining`

Represents the current state of the remaining time duration display of the loaded track.

Range:

Value	Meaning
0	currently showing elapsed time, sets to remaining time
1	currently showing remaining time , sets to elapsed time
2	currently showing both (that means we are showing remaining, set to elapsed

Feedback: None

The Effects Framework

In the list below,

- EffectRack1 leaves room for future expansion to multiple EffectRacks.
- N ranges from 1 to `[EffectRack1],num_effectunits`, inclusive.
- M ranges from 1 to `[EffectRack1_EffectUnitN],num_effectslots`, inclusive. (For a given value of N)
- K ranges from 1 to `[EffectRack1_EffectUnitN_EffectM],num_parameters`, inclusive. (For given values of N and M)
- I ranges from 1 to `[App],num_decks`, inclusive.
- J ranges from 1 to `[App],num_samplers`, inclusive.

New in version 2.0.0.

Linking Values

Effect parameters can be linked to the effect's metaknob. This linkage can be user-controlled by changing the `link_type` and the `link_inverse` control of the parameter. The default link type is loaded from the effect parameter's manifest's `linkHint` property.

Link Type	Integer Value	Interpretation
None	0	Not controlled by the metaknob
Linked	1	Controlled by the metaknob as it is
Linked Left	2	Controlled by the left side of the metaknob
Linked Right	3	Controlled by the right side of the metaknob
Linked Left Right	4	Controlled by both sides of the metaknob

Link Inverse	Integer Value	Interpretation
Normal	0	Linked in equal relation
Inverse	1	Linked in an inverse relation.

EQs and Filters

Equalizers and filters are special effects units. The EQs are controlled by `[EqualizerRack1_[ChannelI]_Effect1]` and the filter knob is controlled by `[QuickEffectRack1_[ChannelI]],super1` and `[QuickEffectRack1_[ChannelI]],enabled`. Users can choose between several options for the effects loaded in these racks in the Equalizers section of the Preferences window.

Controls

[EffectRack1] `num_effectunits`

[EqualizerRack1] `num_effectunits`

[QuickEffectRack1] `num_effectunits`

The number of EffectUnits in this rack

Range: integer, read-only

[EffectRack1_EffectUnitN] `chain_preset_selector`

[EqualizerRack1_[ChannelI]] `chain_preset_selector`

[QuickEffectRack1_[ChannelI]] `chain_preset_selector`

Select EffectChain preset. > 0 goes one forward; < 0 goes one backward.

Range: +1/-1

[EffectRack1_EffectUnitN] `clear`

[EqualizerRack1_[ChannelI]] `clear`

[QuickEffectRack1_[ChannelI]] `clear`

Clear the currently loaded EffectChain in this EffectUnit.

Range: binary

[EffectRack1_EffectUnitN] `enabled`

[EqualizerRack1_[ChannelI]] `enabled`

[QuickEffectRack1_[ChannelI]] `enabled`

If true, the EffectChain in this EffectUnit will be processed. Meant to allow the user a quick toggle for the effect unit.

Range: binary, default true

[EffectRack1_EffectUnitN] `focused_effect`

[EqualizerRack1_[ChannelI]] `focused_effect`

[QuickEffectRack1_[ChannelI]] `focused_effect`

0 indicates no effect is focused; > 0 indicates the index of the focused effect. Focusing an effect only does something if a controller mapping changes how it behaves when an effect is focused.

Range: 0..num_effectslots

[EffectRack1_EffectUnitN] `group_[ChannelI]_enable`

[EqualizerRack1_[ChannelI]] `group_[ChannelI]_enable`

[QuickEffectRack1_[ChannelI]] `group_[ChannelI]_enable`

Whether or not this EffectChain applies to Deck I

Range: binary

[EffectRack1_EffectUnitN] `group_[Headphone]_enable`

Whether or not this EffectChain applies to the Headphone output

Range: binary

[EffectRack1_EffectUnitN] `group_[Master]_enable`

Whether or not this EffectChain applies to the Main output

Range: binary

[EffectRack1_EffectUnitN] `group_[SamplerJ]_enable`

Whether or not this EffectChain applies to Sampler J

Range: binary

[EffectRack1_EffectUnitN] `loaded`

[EqualizerRack1_[ChannelI]] `loaded`

[QuickEffectRack1_[ChannelI]] `loaded`

Whether an EffectChain is loaded into the EffectUnit

Range: binary, read-only

[EffectRack1_EffectUnitN] `loaded_chain_preset`

[EqualizerRack1_[ChannelI]] `loaded_chain_preset`

[QuickEffectRack1_[ChannelI]] `loaded_chain_preset`

0-based index of the currently loaded EffectChain preset. *0* is the empty/passthrough preset, *-1* indicates an unsaved preset (default state of *[EffectRack1_EffectUnitN]*).

Range: integer, -1 .. [`num_chain_presets` - 1]

[EffectRack1_EffectUnitN] `mix`

[EqualizerRack1_[ChannelI]] `mix`

[QuickEffectRack1_[ChannelI]] `mix`

The dry/wet mixing ratio for this EffectChain with the EngineChannels it is mixed with

This is a ControlPotMeter control.

Range: 0.0..1.0

[EffectRack1_EffectUnitN] `next_chain_preset`

[EqualizerRack1_[ChannelI]] `next_chain_preset`

[QuickEffectRack1_[ChannelI]] `next_chain_preset`

Cycle to the next EffectChain preset after the currently loaded preset.

Range: binary

[EffectRack1_EffectUnitN] `num_chain_presets`

[EqualizerRack1_[ChannelI]] `num_chain_presets`

[QuickEffectRack1_[ChannelI]] `num_chain_presets`

The number of effect chain presets available in this EffectUnit, including the empty/passthrough preset “---”.

Range: integer, read-only, >=1

[EffectRack1_EffectUnitN] `num_effectslots`

[EqualizerRack1_[ChannelI]] `num_effectslots`

[QuickEffectRack1_[ChannelI]] `num_effectslots`

The number of effect slots available in this EffectUnit.

Range: integer, read-only

[EffectRack1_EffectUnitN] `prev_chain_preset`

[EqualizerRack1_[ChannelI]] `prev_chain_preset`

[QuickEffectRack1_[ChannelI]] `prev_chain_preset`

Cycle to the previous EffectChain preset before the currently loaded preset.

Range: binary

[EffectRack1_EffectUnitN] `show_focus`

[EqualizerRack1_[ChannelI]] `show_focus`

[QuickEffectRack1_[ChannelI]] `show_focus`

Whether to show focus buttons and draw a border around the focused effect in skins.

This should not be manipulated by skins; it should only be changed by controller mappings.

Range: binary

[EffectRack1_EffectUnitN] `show_parameters`

[EqualizerRack1_[ChannelI]] `show_parameters`

[QuickEffectRack1_[ChannelI]] `show_parameters`

Range: binary

Whether to show all the parameters of each effect in skins or only show metaknobs.

[EffectRack1_EffectUnitN] `super1`

[EqualizerRack1_[ChannelI]] `super1`

[QuickEffectRack1_[ChannelI]] `super1`

The EffectChain superknob. Moves the metaknobs for each effect in the chain.

This is a ControlPotMeter control.

Range: 0.0..1.0

[EffectRack1_EffectUnitN_EffectM] `clear`

[EqualizerRack1_[ChannelI]_Effect1] `clear`

[QuickEffectRack1_[ChannelI]_Effect1] `clear`

Clear the currently loaded Effect in this Effect slot from the EffectUnit.

Range: binary

[EffectRack1_EffectUnitN_EffectM] `effect_selector`

[EqualizerRack1_[ChannelI]_Effect1] `effect_selector`

[QuickEffectRack1_[ChannelI]_Effect1] `effect_selector`

Select Effect – >0 goes one forward, <0 goes one backward.

Range: +1/-1

[EffectRack1_EffectUnitN_EffectM] `enabled`

[EqualizerRack1_[ChannelI]_Effect1] `enabled`

[QuickEffectRack1_[ChannelI]_Effect1] `enabled`

If true, the effect in this slot will be processed. Meant to allow the user a quick toggle for this effect.

Range: binary, default true

[EffectRack1_EffectUnitN_EffectM] `loaded`

[EqualizerRack1_[ChannelI]_Effect1] `loaded`

[QuickEffectRack1_[ChannelI]_Effect1] `loaded`

Whether an Effect is loaded into this EffectSlot

Range: binary, read-only

[EffectRack1_EffectUnitN_EffectM] `loaded_effect`

[EqualizerRack1_[ChannelI]_Effect1] `loaded_effect`

[QuickEffectRack1_[ChannelI]_Effect1] `loaded_effect`

0-based index of the currently loaded effect preset, including the empty/passthrough preset “---”.

Range: integer, 0 .. [`num_effectsavailable` - 1]

[EffectRack1_EffectUnitN_EffectM] `next_effect`

[EqualizerRack1_[ChannelI]_Effect1] `next_effect`

[QuickEffectRack1_[ChannelI]_Effect1] `next_effect`

Cycle to the next effect after the currently loaded effect.

Range: binary

[EffectRack1_EffectUnitN_EffectM] `num_parameters`

[EqualizerRack1_[ChannelI]_Effect1] `num_parameters`

[QuickEffectRack1_[ChannelI]_Effect1] `num_parameters`

The number of parameters the currently loaded effect has.

Range: integer, read-only, 0 if no effect is loaded

[EffectRack1_EffectUnitN_EffectM] `num_parameterslots`

[EqualizerRack1_[ChannelI]_Effect1] `num_parameterslots`

[QuickEffectRack1_[ChannelI]_Effect1] `num_parameterslots`

The number of parameter slots available.

Range: integer, read-only

[EffectRack1_EffectUnitN_EffectM] `num_button_parameters`

[EqualizerRack1_[ChannelI]_Effect1] `num_button_parameters`

[QuickEffectRack1_[ChannelI]_Effect1] `num_button_parameters`

The number of button parameters the currently loaded effect has.

Range: integer, read-only, 0 if no effect is loaded

[EffectRack1_EffectUnitN_EffectM] `num_button_parameterslots`

[EqualizerRack1_[ChannelI]_Effect1] `num_button_parameterslots`

[QuickEffectRack1_[ChannelI]_Effect1] `num_button_parameterslots`

The number of button parameter slots available.

Range: integer, read-only

[EffectRack1_EffectUnitN_EffectM] `meta`

[EqualizerRack1_[ChannelI]_Effect1] `meta`

[QuickEffectRack1_[ChannelI]_Effect1] `meta`

Controls the parameters that are linked to the metaknob.

This is a ControlPotMeter control.

Range: 0..1

[EffectRack1_EffectUnitN_EffectM] `prev_effect`

[EqualizerRack1_[ChannelI]_Effect1] `prev_effect`

[QuickEffectRack1_[ChannelI]_Effect1] `prev_effect`

Cycle to the previous effect before the currently loaded effect.

Range: binary

[EffectRack1_EffectUnitN_EffectM] `parameterK`

[EqualizerRack1_[ChannelI]_Effect1] `parameterK`

[QuickEffectRack1_[ChannelI]_Effect1] `parameterK`

The scaled value of the Kth parameter. See the Parameter Values section for more information.

This is a ControlPotMeter control.

Range: double

[EffectRack1_EffectUnitN_EffectM] `parameterK_link_inverse`

[EqualizerRack1_[ChannelI]_Effect1] `parameterK_link_inverse`

[QuickEffectRack1_[ChannelI]_Effect1] `parameterK_link_inverse`

The link direction of the Kth parameter to the effect's metaknob.

Range: bool

[EffectRack1_EffectUnitN_EffectM] `parameterK_link_type`

[EqualizerRack1_[ChannelI]_Effect1] `parameterK_link_type`

[QuickEffectRack1_[ChannelI]_Effect1] `parameterK_link_type`

The link type of the Kth parameter to the effects's metaknob.

Range: enum

[EffectRack1_EffectUnitN_EffectM] `parameterK_loaded`

[EqualizerRack1_[ChannelI]_Effect1] `parameterK_loaded`

[QuickEffectRack1_[ChannelI]_Effect1] `parameterK_loaded`

Whether or not the Kth parameter slot has an effect parameter loaded into it.

Range: binary, read-only

[EffectRack1_EffectUnitN_EffectM] `parameterK_type`

[EqualizerRack1_[ChannelI]_Effect1] `parameterK_type`

[QuickEffectRack1_[ChannelI]_Effect1] `parameterK_type`

The type of the Kth parameter value. See the Parameter Value Types table.

Range: integer, read-only

[EffectRack1_EffectUnitN_EffectM] `button_parameterK`

[EqualizerRack1_[ChannelI]_Effect1] `button_parameterK`

[QuickEffectRack1_[ChannelI]_Effect1] `button_parameterK`

The value of the Kth parameter. See the Parameter Values section for more information.

Range: double

[EffectRack1_EffectUnitN_EffectM] `button_parameterK_loaded`

[EqualizerRack1_[ChannelI]_Effect1] **`button_parameterK_loaded`**

[QuickEffectRack1_[ChannelI]_Effect1] **`button_parameterK_loaded`**

Whether or not the Kth parameter slot has an effect parameter loaded into it.

Range: binary, read-only

[EffectRack1_EffectUnitN_EffectM] **`button_parameterK_type`**

[EqualizerRack1_[ChannelI]_Effect1] **`button_parameterK_type`**

[QuickEffectRack1_[ChannelI]_Effect1] **`button_parameterK_type`**

The type of the Kth parameter value. See the Parameter Value Types table.

Range: integer, read-only

❗ See also

The `[EffectRack1],show` control can be used to show and hide the effect section in the GUI.

The `[Skin]` group

The `[Skin]` group contains controls that are used to selective show and hide parts of the graphical user interface of Mixxx to suit your needs.

❗ Note

Some skins may not support some or all of these controls. In this case, triggering the unsupported control will do nothing.

[Skin] **`show_effectrack`**

Toggle the display of the effect rack in the user interface.

Range: binary
Feedback: Effect rack is shown/hidden.

New in version 2.4.0: Replaces the deprecated `[EffectRack1],show` control.

[Skin] **`show_library_coverart`**

Toggle the display of cover art in the library section of the user interface.

Range: binary
Feedback: Cover art in the library is shown/hidden.

New in version 2.4.0: Replaces the deprecated `[Library],show_coverart` control.

[Skin] `show_maximized_library`

Toggle maximized view of library section of the user interface.

Range: binary

Feedback: The library section of the user interface is enlarged/shrunk.

New in version 2.4.0: Replaces the deprecated `[Master],maximize_library` control.

[Skin] `show_samplers`

Toggle the display of sampler banks in the user interface.

Range: binary

Feedback: Sampler banks are shown/hidden.

New in version 2.4.0: Replaces the deprecated `[Samplers],show_samplers` control.

[Skin] `show_vinylcontrol`

Toggle the vinyl control section in the user interface.

Range: binary

Feedback: Vinyl controls are shown/hidden.

New in version 2.4.0: Replaces the deprecated `[VinylControl],show_vinylcontrol` control.

Deprecated controls

These controls have been deprecated and may be removed in a future version of Mixxx. In the meantime, skins and controller mappings that still use them will keep working, but using the suggested alternatives is strongly recommended.

[Master] `num_decks`

The number of decks currently enabled.

Range: integer

Feedback: None

New in version 1.9.0.

Deprecated since version 2.4.0: Use `[App],num_decks` instead.

[Master] `num_samplers`

The number of samplers currently enabled.

Range:	integer
Feedback:	None

New in version 1.9.0.

Deprecated since version 2.4.0: Use `[App],num_samplers` instead.

[Master] `num_preview_decks`

The number of preview decks currently enabled.

Range:	integer
Feedback:	None

New in version 1.9.0.

Deprecated since version 2.4.0: Use `[App],num_preview_decks` instead.

[Master] `num_microphones`

The number of microphone inputs that can be configured.

Range:	integer
Feedback:	None

New in version 2.2.4.

Deprecated since version 2.4.0: Use `[App],num_microphones` instead.

[Master] `num_auxiliaries`

The number of auxiliary inputs that can be configured.

Range:	integer
Feedback:	None

New in version 2.2.4.

Deprecated since version 2.4.0: Use `[App],num_auxiliaries` instead.

[Master] `samplerate`

The current output sample rate (default: 44100 Hz).

Range:	absolute value (in Hz)
Feedback:	None *Deprecated since version 2.4.0:* Use `[App],samplerate` instead.

[Master] `headVolume`

Adjust headphone volume.

Range:	0.0..1.0..5.0
Feedback:	Headphone Gain knob

Deprecated since version 2.0.0: Use `[Master],headGain` instead.

[Master] `volume`

Adjust main volume.

Range:	0.0..1.0..5.0
Feedback:	Main Gain knob

Deprecated since version 2.0.0: Use `[Master],gain` instead.

[Master] `maximize_library`

Toggle maximized view of library.

Range:	binary
Feedback:	Toggle maximized view of library

New in version 2.0.0.

Deprecated since version 2.4.0: Use `[Skin],show_maximized_library` instead.

[Samplers] `show_samplers`

Range:	binary
Feedback:	Shows Sampler bank(s)

Deprecated since version 2.4.0: Use `[Skin],show_samplers` instead.

[VinylControl] `show_vinylcontrol`

Toggle the vinyl control section in skins.

Range:	binary
Feedback:	Vinyl controls are shown

New in version 1.10.0.

Deprecated since version 2.4.0: Use `[Skin],show_vinylcontrol` instead.

[Library] `show_coverart`

Toggle the Cover Art in Library

Range:	Binary

Deprecated since version 2.4.0: Use `[Skin],show_library_coverart` instead.

[EffectRack1] `show`

Show the Effect Rack

Range: binary

Deprecated since version 2.4.0: Use `[Skin],show_effectrack` instead.

[ChannelN] `hotcue_X_enabled`

[PreviewDeckN] `hotcue_X_enabled`

[SamplerN] `hotcue_X_enabled`

Indicates if hotcue slot X is set, active or empty.

Range (read-only):

Value	Meaning
0	Hotcue X is not set
1	Hotcue X is set
2	Hotcue X is active (saved loop is enabled or hotcue is previewing)

New in version 1.8.0.

Changed in version 2.4.0: Added support for “active” state.

Deprecated since version 2.4.0: Use `[ChannelN],hotcue_X_status` instead.

[ChannelN] `sync_master`

[PreviewDeckN] `sync_master`

[SamplerN] `sync_master`

Sets deck as leader clock.

Range: binary

Feedback: If enabled, the `Sync` button stays lit and tempo slider snap to the appropriate value. Slider adjustments are linked on all decks that have sync lock enabled.

New in version 2.0.0.

Changed in version 2.3.0: This button just enables sync lock mode (similar to `[ChannelN],sync_enabled`), it does not actually guarantee the deck will be the sync leader. This will be fixed in a future version. .. deprecated:: 2.4.0 Use `[ChannelN],sync_leader`, `[PreviewDeckN],sync_leader` and `[SamplerN],sync_leader` instead.

[ChannelN] **`beatloop`**

[PreviewDeckN] **`beatloop`**

[SamplerN] **`beatloop`**

Setup a loop over the set number of beats.

Range:	positive real number
Feedback:	A loop is shown over the set number of beats.

Deprecated since version 2.1.0: Use `[ChannelN],beatloop_size` and *[ChannelN],beatloop_toggle* instead.

[ChannelN] **`reloop_exit`**

[PreviewDeckN] **`reloop_exit`**

[SamplerN] **`reloop_exit`**

Toggles the current loop on or off. If the loop is ahead of the current play position, the track will keep playing normally until it reaches the loop.

Range:	binary
Feedback:	Loop range in waveform activates or deactivates.

Deprecated since version 2.1.0: Use `[ChannelN],reloop_toggle` instead.

[ChannelN] **`jog`**

[PreviewDeckN] **`jog`**

[SamplerN] **`jog`**

Affects relative playback speed and direction for short instances (additive & is automatically reset to 0).

Range:	-3.0..3.0
Feedback:	waveform

Deprecated since version ??: Use the JavaScript `engine.scratch` functions instead.

[ChannelN] **`scratch`**

[PreviewDeckN] **`scratch`**

[SamplerN] **`scratch`**

Affects playback speed and direction (differently whether currently playing or not) (multiplicative).

Range: -3.0..3.0
Feedback: Waveform

Deprecated since version ??: Use the JavaScript `engine.scratch` functions instead.

[ChannelN] **`filter`**

[PreviewDeckN] **`filter`**

[SamplerN] **`filter`**

Toggles the filter effect.

Range: binary
Feedback: Filter button

New in version 2.0.0.

Deprecated since version 2.0.0: Use `[QuickEffectRack1_[ChannelN]_Effect1],enabled` instead.

[ChannelN] **`filterDepth`**

[PreviewDeckN] **`filterDepth`**

[SamplerN] **`filterDepth`**

Adjusts the intensity of the filter effect.

Range: default
Feedback: Filter depth knob

New in version 2.0.0.

Deprecated since version 2.0.0: Use `[QuickEffectRack1_[ChannelN]],super1` instead.

[ChannelN] **`filterLow`**

[PreviewDeckN] **`filterLow`**

[SamplerN] **`filterLow`**

Adjusts the gain of the low EQ filter.

Range: 0.0..1.0..4.0
Feedback: Low EQ knob

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelN]_Effect1],parameter1` instead.

[ChannelN] **`filterLowKill`**

[PreviewDeckN] **`filterLowKill`**

[SamplerN] **`filterLowKill`**

Holds the gain of the low EQ to -inf while active

Range:	binary
Feedback:	Low EQ kill switch

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelI]_Effect1],button_parameter1` instead.

[ChannelN] **`filterMid`**

[PreviewDeckN] **`filterMid`**

[SamplerN] **`filterMid`**

Adjusts the gain of the mid EQ filter..

Range:	0.0..1.0..4.0
Feedback:	Mid EQ knob

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelI]_Effect1],parameter2` instead.

[ChannelN] **`filterMidKill`**

[PreviewDeckN] **`filterMidKill`**

[SamplerN] **`filterMidKill`**

Holds the gain of the mid EQ to -inf while active.

Range:	binary
Feedback:	Mid EQ kill switch

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelI]_Effect1],button_parameter2` instead.

[ChannelN] **`filterHigh`**

[PreviewDeckN] **`filterHigh`**

[SamplerN] **`filterHigh`**

Adjusts the gain of the high EQ filter.

Range:	0.0..1.0..4.0

Feedback: High EQ knob

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelI]_Effect1],parameter3` instead.

[ChannelN] `filterHighKill`

[PreviewDeckN] `filterHighKill`

[SamplerN] `filterHighKill`

Holds the gain of the high EQ to -inf while active.

Range: binary

Feedback: High EQ kill switch

Deprecated since version 2.0.0: Use `[EqualizerRack1_[ChannelI]_Effect1],button_parameter3` instead.

[ChannelN] `beatloop_X`

[PreviewDeckN] `beatloop_X`

[SamplerN] `beatloop_X`

Setup a loop over X beats. A control exists for X = 0.03125, 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64

Range: toggle

Feedback: A loop is shown over X beats.

New in version 1.10.0.

Deprecated since version 2.0.0: Use `[ChannelN],beatloop_X_activate` instead.

[MicrophoneN] `enabled`

[AuxiliaryN] `enabled`

1 if a channel input is enabled, 0 if not.

Range: binary

Feedback: Microphone is enabled.

New in version 1.10.0.

Deprecated since version 2.0.0: Use `[MicrophoneN],input_configured` instead.

[MicrophoneN] `master`

[AuxiliaryN] `master`

Hold value at 1 to mix channel input into the main output. For `[MicrophoneN]` use `[MicrophoneN],talkover` instead. Note that `[AuxiliaryN]` also take `[AuxiliaryN],orientation` into account.

Range:	binary
Feedback:	Auxiliary: Play button Microphone: N/A

Deprecated since version 2.4.0: Use `[MicrophoneN],talkover` and `[AuxiliaryN],main_mix` instead.

[MicrophoneN] `orientation`

New in version 1.10.0.

Deprecated since version 1.10.0: The control is not processed in the Mixer, which is also why there are no orientation controls for Microphones in the GUI.

[Playlist] `LoadSelectedIntoFirstStopped`

Performs the same action action like `[Library],GoToItem` does when the tracks table has focus, just regardless of the focus.

Deprecated since version 2.1.0: Use `[Library],GoToItem` instead.

[Playlist] `SelectNextPlaylist`

Switches to the next view (Library, Queue, etc.)

Deprecated since version 2.1.0: Use `[Library],MoveDown` instead.

[Playlist] `SelectPrevPlaylist`

range:	binary
feedback:	Library sidebar

Switches to the previous view (Library, Queue, etc.)

> *Deprecated since version 2.1.0:* Use `[Library],MoveUp` instead.

[Playlist] `ToggleSelectedSidebarItem`

Toggles (expands/collapses) the currently selected sidebar item.

New in version 1.11.0.

Deprecated since version 2.1.0: Use `[Library],GoToItem` instead.

[Playlist] `SelectNextTrack`

Scrolls to the next track in the track table.

Deprecated since version 2.1.0: Use `[Library],MoveDown` instead.

[Playlist] **`SelectPrevTrack`**

Scrolls to the previous track in the track table.

Deprecated since version 2.1.0: Use `[Library],MoveUp` instead.

[EffectRack1_EffectUnitN] **`next_chain`**

[EqualizerRack1_[ChannelI]] **`next_chain`**

[QuickEffectRack1_[ChannelI]] **`next_chain`**

Cycle to the next EffectChain preset after the currently loaded preset.

Deprecated since version 2.4.0: Use `[EffectRack1_EffectUnitN],next_chain_preset` instead.

[EffectRack1_EffectUnitN] **`prev_chain`**

[EqualizerRack1_[ChannelI]] **`prev_chain`**

[QuickEffectRack1_[ChannelI]] **`prev_chain`**

Cycle to the next EffectChain preset after the currently loaded preset.

Deprecated since version 2.4.0: Use `[EffectRack1_EffectUnitN],prev_chain_preset` instead.

[EffectRack1_EffectUnitN] **`chain_selector`**

[EqualizerRack1_[ChannelI]] **`chain_selector`**

[QuickEffectRack1_[ChannelI]] **`chain_selector`**

Select EffectChain preset. > 0 goes one forward; < 0 goes one backward.

Deprecated since version 2.4.0: Use `[EffectRack1_EffectUnitN],chain_preset_selector` instead.

Removed controls

These controls have been removed from Mixxx. Skins and controller mappings that attempt to use them will not work correctly.

[ChannelN] **`flanger`**

Toggles the flanger effect.

Deprecated since version 2.0.0: This control has been **removed** without a direct replacement. Use the effects framework instead.

[ChannelN] `Hercules1`

[ChannelN] `Hercules2`

[ChannelN] `Hercules3`

[ChannelN] `Hercules4`

Deprecated since version ??: This control has been **removed**.

[ChannelN] `NextTask`

Deprecated since version ??: This control has been **removed**.

[ChannelN] `NextTrack`

Deprecated since version ??: This control has been **removed**.

[ChannelN] `PrevTask`

Deprecated since version ??: This control has been **removed**.

[ChannelN] `PrevTrack`

Deprecated since version ??: This control has been **removed**.

[ChannelN] `transform`

Deprecated since version ??: This control has been **removed**.

[Flanger] `lfoDepth`

Adjusts the intensity of the flange effect

Deprecated since version 2.0.0: This control has been **removed** without a direct replacement. Use the effects framework instead.

[Flanger] `lfoDelay`

Adjusts the phase delay of the flange effect in microseconds

Deprecated since version 2.0.0: This control has been **removed** without a direct replacement. Use the effects framework instead.

[Flanger] `lfoPeriod`

Adjusts the wavelength of the flange effect in microseconds

Deprecated since version 2.0.0: This control has been **removed** without a direct replacement. Use the effects framework instead.

[EffectRack1] `clear`

[EqualizerRack1] `clear`

[QuickEffectRack1] `clear`

Clear the Effect Rack

Deprecated since version 2.4.0: This control has been **removed** without a direct replacement. Use the effects framework instead.