

Advanced Fader Range Setup

This is an example of an eventhandler that makes it so moving the full range of the physical fader only moved the top half of the actual "fader" parameter, using the "Analog to Analog" conversion and Map1:

Routes:

InputMapping

Default

Create new...

Route: Default

Repeat settings:

Threshold

Repeat Thresholds

For Analog Outputs:

Map1:

Input Start

Input End 1000

Output Start 500

Output End 1000

Map2:

Input Start

Input End

Output Start

Output End

Cancel Submit

```
"EventHandlers": {
  "change": {
    "EventPreProc": {
      "A2A": {
        "InputMapping": {
          "Default": {
            "Map1": {
              "InputEnd": 1000,
              "OutputStart": 500,
              "OutputEnd": 1000
            }
          }
        }
      }
    }
  }
}
```

```

    },
    "BinarySetValues": {},
    "IOReference": {}
  }
},

```

This does however completely remove the option to go to 0 on the param, if you want that you might want this kind of code instead where Map2 makes it so the last 5% of the physical fader makes it jump down to 0 or "inf" when talking audio faders, while still keeping the scaling on the rest of the fader positions:

Routes: InputMapping

Default

Create new...

Route: Default

Repeat settings:

Threshold

☐ Repeat Thresholds

For Analog Outputs:

Map1:

Input Start

Input End

Output Start

Output End

Map2:

Input Start: 50

Input End: 1000

Output Start: 500

Output End: 1000

Cancel Submit

```

"EventHandlers": {
  "change": {
    "EventPreProc": {
      "A2A": {
        "InputMapping": {
          "Default": {
            "Map1": {},
            "Map2": {
              "InputStart": 50,
              "InputEnd": 1000,
              "OutputStart": 500,
              "OutputEnd": 1000
            }
          }
        }
      }
    }
  }
}

```

```
    },  
    "BinarySetValues": {},  
    "IOReference": {}  
  }  
},
```

All Values here are set in a "Normalized" range, so this will always be a number between 0 and 1000, think of it as a % of the values on your param where 1000 = 100%, 500 = 50% and so on.

InputStart and InputEnd controls the max movement of the physical fader, so setting these can give you "deadzones" at the ends of a fader, if you'r newer quite able to hit the max and min values.

by default you would always want them on 0 or 1000 on a fader when using map1.
If you're using map2 you might want the lower one to be a certain % above 0, so our recommendation would be to start at 50 (5%) or 100 (10%) and experiment from there, and keeping the max on 1000.

OutputStart is the minimum value you want the fader to set
OutputEnd is the maximum value you want the fader to set

In the example above you will see them limit the output values to only cover the upper 50% of the parameter by using the values 500 and 1000. play around with these to fit your desired outcome range.

Feedback

The above section only covers how to configure the eventhandler. For getting correct feedback for motorfaders, an additional step is needed.

Open the default feedback handler, and open the advanced feedback options. Select a Static Value Mapping Type of 'Coords'. Now add the Static Value Mapping Coords graph. Open the Coords popup and press '+' to add a new point. Set this new point to the same mapping as the eventhandler map. This might take some experimenting.

The Json should look something like this:

```
"FeedbackDefault": {  
  "Extended": {  
    "IOReference": {},  
    "StaticValueMapping": {  
      "Type": "Coords",  
      "CoordPairs": [  
        ]
```

```
    0,  
    0  
  ],  
  [  
    900,  
    1000  
  ],  
  [  
    1000,  
    1000  
  ]  
]  
}  
}  
}
```

Revision #3

Created 30 January 2023 07:54:25 by Andreas Hauge Thomsen

Updated 17 January 2025 11:29:08 by Samuel Jakobsen