

Modulation Source Types

Description

Enumerator to identify the different modulation sources.

Available in: Controller.

Index	source	sourceInfo1	sourceInfo2	Comment
0	ModulationSource.unassigned	-	-	
1	ModulationSource.lfo1	-	-	
2	ModulationSource.lfo2	-	-	
3	ModulationSource.ampEnv	-	-	
4	ModulationSource.filterEnv	-	-	
5	ModulationSource.pitchEnv	-	-	
6	ModulationSource.userEnv	-	-	
7	ModulationSource.stepMod	-	-	
8	ModulationSource.glide	-	-	
9	ModulationSource.keyFollow	-	-	
10	ModulationSource.noteOnVelocity	-	-	
11	ModulationSource.noteOnVelocitySquared	-	-	
12	ModulationSource.noteOnVelocityNormalized	-	-	
13	ModulationSource.noteOffVelocity	-	-	
14	ModulationSource.pitchBend	-	-	
15	ModulationSource.modWheel	-	-	
16	ModulationSource.aftertouch	-	-	
17	ModulationSource.midiControl	MIDI controller number (0 - 127)	-	
18	ModulationSource.quickControl	Element object of the layer	Index of the quick control (1 - 11)	
19	ModulationSource.modulationModule	Element object of the MIDI module	Number of the output channel	
20	ModulationSource.noteExpression	Number of the custom note expression (1-8)	-	
21	ModulationSource.noise	-	-	
22	ModulationSource.output	-	-	
23	ModulationSource.bus1	-	-	
24	ModulationSource.bus2	-	-	
25	ModulationSource.bus3	-	-	
26	ModulationSource.bus4	-	-	
27	ModulationSource.bus5	-	-	
28	ModulationSource.bus6	-	-	
29	ModulationSource.bus7	-	-	
30	ModulationSource.bus8	-	-	
31	ModulationSource.bus9	-	-	
32	ModulationSource.bus10	-	-	
33	ModulationSource.bus11	-	-	
34	ModulationSource.bus12	-	-	
35	ModulationSource.bus13	-	-	
36	ModulationSource.bus14	-	-	
37	ModulationSource.bus15	-	-	
38	ModulationSource.bus16	-	-	
39	-	-	-	

40	-	-	-	
41	ModulationSource.sampleAndHold	Index of the S&H (0 - 5)	-	Source 2 only

Example

```

-- define the modulation sources and infos in an array
modSources = {
  { source = ModulationSource.lf01, bipolar = 1 },
  { source = ModulationSource.midiControl, bipolar = 0, sourceInfo1 = 1 },
  { source = ModulationSource.quickControl, bipolar = 1, sourceInfo1 = this.program, sourceInfo2 = 1 },
  { source = ModulationSource.modulationModule, bipolar = 0, sourceInfo1 = this, sourceInfo2 = 1 },
  { source = ModulationSource.modulationModule, bipolar = 0, sourceInfo1 = this, sourceInfo2 = 2 },
  { source = ModulationSource.noteExpression, bipolar = 0, sourceInfo1 = 1 }
}

-- define two modulation outputs for the script module
defineModulation("50%", false)
defineModulation("100%", false)

-- calculate the modulation outputs
function calcModulation()
  return 0.5, 1
end

-- get the element object of the first zone
zone = this.program:findZones(true)[1]

-- assign the modulation sources to source 1 in the modulation rows 1 to 6
for i=1, #modSources do
  local modRow = zone:getModulationMatrixRow(i)
  modRow:setSource1(modSources[i].source, modSources[i].sourceInfo1, modSources[i].sourceInfo2)
  modRow:setParameter("Source1.Polarity", modSources[i].bipolar) -- set the default polarity of the source
end

-- assign the sample & hold to source 2 in modulation row 1
modRow = zone:getModulationMatrixRow(1)
modRow:setSource2(ModulationSource.sampleAndHold, 0)

```

See Also: [ModulationMatrixRow](#), [getModulationMatrixRow](#), [setSource1](#), [setSource2](#), [getSource1](#), [getSource2](#), [Modulation Destination Types](#)