

Zone

The Zone class inherits all properties and methods of the [Element](#) class.

On this page:

[Zone Class](#), [Zone Constructor](#), [getModulationMatrixRow](#), [getOutputBus](#), [setOutputBus](#)

Element

[Element Class](#), [findChildren](#), [getChild](#), [getParameter](#), [getParameterDefinition](#), [getParameterNormalized](#), [hasParameter](#), [removeFromParent](#), [setName](#), [setParameter](#), [setParameterNormalized](#)

Class Hierarchy

- [AudioFile](#)
- [Element](#)
 - [Bus](#)
 - [Effect](#)
 - [Instance](#)
 - [Layer](#)
 - [Program](#)
 - [MidiModule](#)
 - [ModulationMatrixRow](#)
 - [Slot](#)
 - [Zone](#)
- [Event](#)
- [LoadProgress](#)
- [ParameterDefinition](#)

Classes

Zone Class

Description

The [Element](#) object of a zone can be obtained with [findZones](#) or [getZone](#). It has the following fields.

Available in: Controller, Processor.

Fields

.keyLow	The lowest key of the zone.	number
.keyHigh	The highest key of the zone.	number
.velLow	The lowest velocity of the zone.	number
.velHigh	The highest velocity of the zone.	number

Example

```
-- print the key and velocity range of the first zone in the program
zone = this.program.findZones(true)[1]
print(zone.keyLow)
print(zone.keyHigh)
print(zone.velLow)
print(zone.velHigh)
```

[Jump to Top](#)

Constructors

Zone Constructor

Zone()

Description

Constructor to create a new [Zone](#) object.

Available in: Controller.

Return Values

Returns a new [Zone](#) object.

Example

```
-- This function creates different types of objects in the Program Tree.
-- The objects in the Program Tree do not have a name. You will see only their icons.
function createProgram()
  local inst = this.program.instance
  local prg = Program()
  local bus = Bus()
  prg:appendBus(bus)
  inst:setProgram(prg, 1)
  local layer = Layer()
  prg:appendLayer(layer)
  layer:appendZone(Zone())
  local mm = MidiModule('MIDI Player')
  layer:appendMidiModule(mm)
  local fx = Effect('Distortion')
  bus:appendEffect(fx)
end

createProgram()
```

[Jump to Top](#)

Methods

getModulationMatrixRow

`getModulationMatrixRow(rowNumber)`

Description

Function to obtain the [ModulationMatrixRow](#) object of the specified modulation matrix row. The modulation matrix row is determined by the [Zone](#) object of the zone and the index of the modulation matrix row.

Available in: Controller, Processor.

Arguments

rowNumber	The index of the modulation matrix row in the range from 1 to 32.	number
------------------	---	--------

Return Values

The [ModulationMatrixRow](#) object of the specified modulation matrix row.

Example

```
-- get the element object of the first zone in the program
zone = this.program:findZones(true)[1]
-- get the element object of the first modulation matrix row
modRow = zone:getModulationMatrixRow(1)
-- print the row number of the specified modulation matrix row
print(modRow.rowNumber)
```

[Jump to Top](#)

getOutputBus

`getOutputBus()`

Description

Function to retrieve the currently assigned output bus of a zone or bus.

Available in: Controller, Processor.

Return Values

Returns the [Bus](#) object of the currently assigned output bus or `nil` if the default routing is used.

Example

```
-- raise an error if no output bus is assigned
zone = this.parent:getZone()
assert(zone:getOutputBus(), "No output bus assigned. The default routing is used!")
```

[Jump to Top](#)

setOutputBus

`setOutputBus(bus)`

Description

Function to assign the output of a zone or bus to the specified output bus. The sending zone or bus is determined by its [Element](#) object. The receiving output bus is specified by its [Bus](#) object. Setting the output bus to `nil` enables the default signal routing for the zone or bus.

Output busses that are higher up in the hierarchy of the **Program Tree** can be assigned freely. If the sending bus and the receiving output bus have the same parent layer, the output bus must come later in the signal flow.

Available in: Controller.

Arguments

bus	The Bus object of the bus that you want to assign, or <code>nil</code> . Bus or <code>nil</code>
------------	--

Example

```
-- assign the output of the zone to the Master output bus of the plug-in
zone = this.parent:getZone()
masterbus = this.program.instance:getBus(1)

zone:setOutputBus(masterbus)

print("Output of "..zone.name.." is assigned to "..masterbus.name..")
```

[Jump to Top](#)