
Midistream Documentation

Release 0.1

b3b

Jan 22, 2021

Contents

1	Related resources	3
2	Quick start development environment	5
3	Build	7
4	Examples	9
5	API	11
5.1	Midi	11
5.2	Helpers	12
6	Usage Example	15
	Python Module Index	17
	Index	19

Package to synthesize and playback MIDI commands from [Python for Android](#).

Midistream is a wrapper for the Bill Farmer [Midi Driver](#), and includes *libmidi.so* libraries from the MidiDriver build.

[Previous](#) version was using system version of Sonivox EAS library and [Audiostream](#) for playback.

Generated documentation: <https://midistream.readthedocs.org>

CHAPTER 1

Related resources

- [EAS documentation](#)
- [Android Issue 8201: Add a real-time MIDI API for Sonivox synthesizer](#)
- [Android MIDI driver using Sonivox EAS library](#)

CHAPTER 2

Quick start development environment

midistream is included in [PythonHere](#) app, together with the [Jupyter Notebook](#) it could be used as a development environment.

Usage examples: <https://herethere.me/examples/midi.html>

CHAPTER 3

Build

The following instructions are for building app with [buildozer](#) tool.

buildozer.spec requirements should include *midistream* and *mididriver*, path to *midistream* recipes directory should be set:

```
requirements =
    mididriver,
    https://github.com/b3b/midistream/archive/master.zip,
p4a.local_recipes = /path/to/cloned/repo/recipes
```

App configuration example: [buildozer.spec](#)

CHAPTER 4

Examples

See examples/ directory.

examples/instrument could be build with *buildozer*:

```
cd examples/instrument
buildozer android debug deploy run logcat
```


5.1 Midi

exception `midistream.MIDIException`
MIDI error.

class `midistream.ReverbPreset`
Parameter settings for reverb effect.

OFF = -1
Reverb effect off

LARGE_HALL = 0
Large hall preset

HALL = 1
Hall preset

CHAMBER = 2
Chamber preset

ROOM = 3
Room preset

class `midistream.Synthesizer`
MIDI Synthesizer.

config
Synthesizer configuration dictionary.

volume
Master volume in dB, 100 is max.

Getter Returns the volume for the mix engine.

Setter Set the master volume for the mix engine.

reverb

Reverb effect preset.

Getter Returns currently used *ReverbPreset*.

Setter Set *ReverbPreset* to use.

close()

Stop MIDI rendering and playback.

write(data: AnyStr)

Write MIDI commands to synthesizer stream.

5.2 Helpers

Helpers to work with MIDI messages.

`midistream.helpers.midi_note_on(note: int, channel: int = 0, velocity: int = 64) → List[int]`
MIDI 9nH message - note on.

```
>>> midi_note_on(70)
[144, 70, 64]
>>> midi_note_on(70, velocity=127, channel=15)
[159, 70, 127]
```

`midistream.helpers.midi_note_off(note: int, channel: int = 0, velocity: int = 0) → List[int]`
MIDI 8nH message - note off.

```
>>> midi_note_off(70)
[128, 70, 0]
>>> midi_note_off(70, channel=15)
[143, 70, 0]
```

`midistream.helpers.midi_program_change(program: int, channel: int = 0) → List[int]`
MIDI CnH message - program change.

```
>>> midi_program_change(80, 1)
[193, 80]
```

`midistream.helpers.midi_control_change(controller: int, value: int = 0, channel: int = 0) → List[int]`
MIDI BnH message - control change.

```
>>> midi_control_change(7, value=127, channel=1)
[177, 7, 127]
```

`midistream.helpers.midi_command_increase_channel(command: List[int], inc: int) → List[int]`
Increase channel number of a given command.

```
>>> command = [177, 7, 127]
>>> midi_command_increase_channel(command, -7)
[170, 7, 127]
>>> command
[177, 7, 127]
```

class midistream.helpers.Control

Control function number for Control Change messages.

See: <https://www.midi.org/specifications-old/item/table-3-control-change-messages-data-bytes-2>

modulation = 1
Modulation Wheel

volume = 7
Channel Volume

pan = 10
Pan

all_sound_off = 120
All Sound Off

`midistream.helpers.midi_channels()` → Generator[int, None, None]
Generator of MIDI channels numbers, with percussion (9) channel omitted.

```
>>> list(midi_channels())
[0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15]
```

`midistream.helpers.midi_notes = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, ...]`
All MIDI notes list (from 0 to 127)

`midistream.helpers.note_name(note: int) → str`
Returns name with accidental and octave number for a given note number.

```
>>> note_name(60)
'C4'
>>> note_name(90)
'Fs6'
```

`midistream.helpers.parse_note(text: str) → int`
Parse note number from text.

Parameters `text` – [Note name](optional: “s” - sharp, “b” - flat)[octave number]

Raises `ValueError`

```
>>> parse_note("C4")
60
>>> parse_note("Cs4")
61
>>> parse_note("Cb4")
59
```

class `midistream.helpers.Note`
Note number.

```
>>> Note.A0
21
>>> Note.As0
22
>>> Note.Ab0
20
>>> Note.G9
127
```

`midistream.helpers.midi_instruments = {0: 'Acoustic Grand Piano', 1: 'Bright Acoustic Pi`
MIDI instruments number => name dictionary

CHAPTER 6

Usage Example

```
>>> from midistream import ReverbPreset, Synthesizer
>>> midi = Synthesizer()
>>> midi.config
{'libVersion': 50727438, 'checkedVersion': 0, 'ma3xVoices': 64, 'numChannels': 2,
 ↪ 'sampleRate': 22050, 'mixBufferSize': 128, 'filterEnabled': 1, 'buildTimeStamp': ↪
 ↪ 1195621085, 'buildGUID': b'1feda229-b9a8-45e9-96f4-73c0a80e7220'}
>>> midi.volume
90
>>> midi.volume = 70 # Set master volume
>>> midi.reverb = ReverbPreset.LARGE_HALL # Enable reverb effect
>>> midi.write([0x90, 60, 127]) # On middle C note with maximum velocity
>>> import time ; time.sleep(2)
>>> midi.write([0x80, 60, 127]) # Off middle C note with maximum velocity
# Using helpers
>>> from midistream.helpers import Note, midi_note_on
>>> midi.write(midi_note_on(Note.C4) + midi_note_on(Note.Es5))
```


m

`midistream`, [11](#)

`midistream.helpers`, [12](#)

A

`all_sound_off` (*midistream.helpers.Control attribute*), 13

C

`CHAMBER` (*midistream.ReverbPreset attribute*), 11
`close()` (*midistream.Synthesizer method*), 12
`config` (*midistream.Synthesizer attribute*), 11
`Control` (*class in midistream.helpers*), 12

H

`HALL` (*midistream.ReverbPreset attribute*), 11

L

`LARGE_HALL` (*midistream.ReverbPreset attribute*), 11

M

`midi_channels()` (*in module midistream.helpers*), 13
`midi_command_increase_channel()` (*in module midistream.helpers*), 12
`midi_control_change()` (*in module midistream.helpers*), 12
`midi_instruments` (*in module midistream.helpers*), 13
`midi_note_off()` (*in module midistream.helpers*), 12
`midi_note_on()` (*in module midistream.helpers*), 12
`midi_notes` (*in module midistream.helpers*), 13
`midi_program_change()` (*in module midistream.helpers*), 12
`MIDIException`, 11
`midistream` (*module*), 11
`midistream.helpers` (*module*), 12
`modulation` (*midistream.helpers.Control attribute*), 13

N

`Note` (*class in midistream.helpers*), 13
`note_name()` (*in module midistream.helpers*), 13

O

`OFF` (*midistream.ReverbPreset attribute*), 11

P

`pan` (*midistream.helpers.Control attribute*), 13
`parse_note()` (*in module midistream.helpers*), 13

R

`reverb` (*midistream.Synthesizer attribute*), 11
`ReverbPreset` (*class in midistream*), 11
`ROOM` (*midistream.ReverbPreset attribute*), 11

S

`Synthesizer` (*class in midistream*), 11

V

`volume` (*midistream.helpers.Control attribute*), 13
`volume` (*midistream.Synthesizer attribute*), 11

W

`write()` (*midistream.Synthesizer method*), 12