Previous topic
gnuradio.pmt
Next topic
gnuradio.analog
This Page
Show Source
Quick search
Go
Enter search terms or a module, class or function name.

gnuradio.audio

Blocks to connect to audio sources (mic-in) and sinks (speaker-out) ports on a computer.

The underlying hardware driver is system and OS dependent and this module should automatically discover the correct one to use.

gnuradio.audio.**source**(*int sampling_rate*, *std::string const device_name*, *bool ok_to_block=True*) → source_sptr

Creates a source from an audio device.

Constructor Specific Documentation:

Creates a source from an audio device at a specified sample_rate. The specific audio device to use can be specified as the device_name parameter. Typical choices are:

- pulse, hw:0,0, plughw:0,0, surround51, /dev/dsp

Parameters:
- **sampling_rate** –
- **device_name** –
- **ok_to_block** –

source_sptr.**active_thread_priority**(*source_sptr self*) → int

source_sptr.**declare_sample_delay**(*source_sptr self*, *int which*, *int delay*)

declare_sample_delay(source_sptr self, unsigned int delay)

source_sptr.**message_subscribers**(*source_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

source_sptr.**min_noutput_items**(*source_sptr self*) → int

source_sptr.**pc_input_buffers_full_avg**(*source_sptr self*, *int which*) → float

pc_input_buffers_full_avg(source_sptr self) -> pmt_vector_float

source_sptr.**pc_noutput_items_avg**(*source_sptr self*) → float

source_sptr.**pc_nproduced_avg**(*source_sptr self*) → float

source_sptr.**pc_output_buffers_full_avg**(*source_sptr self*, *int which*) → float

pc_output_buffers_full_avg(source_sptr self) -> pmt_vector_float

source_sptr.**pc_throughput_avg**(*source_sptr self*) → float

source_sptr.**pc_work_time_avg**(*source_sptr self*) → float

source_sptr.**pc_work_time_total**(*source_sptr self*) → float

source_sptr.**sample_delay**(*source_sptr self*, *int which*) → unsigned int

source_sptr.**set_min_noutput_items**(*source_sptr self*, *int m*)

source_sptr.**set_thread_priority**(*source_sptr self*, *int priority*) → int

source_sptr.**thread_priority**(*source_sptr self*) → int

gnuradio.audio.**sink**(*int sampling_rate*, *std::string const device_name*, *bool ok_to_block=True*) → sink_sptr

Creates a sink from an audio device.

Constructor Specific Documentation:

Creates a sink from an audio device at a specified sample_rate. The specific audio device to use can be specified as the device_name parameter. Typical choices are:

- pulse, hw:0,0, plughw:0,0, surround51, /dev/dsp

Parameters:
- **sampling_rate** –
- **device_name** –
- **ok_to_block** –

sink_sptr.**active_thread_priority**(*sink_sptr self*) → int

sink_sptr.**declare_sample_delay**(*sink_sptr self*, *int which*, *int delay*)
declare_sample_delay(sink_sptr self, unsigned int delay)

sink_sptr.**message_subscribers**(*sink_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

sink_sptr.**min_noutput_items**(*sink_sptr self*) → int

sink_sptr.**pc_input_buffers_full_avg**(*sink_sptr self*, *int which*) → float
pc_input_buffers_full_avg(sink_sptr self) -> pmt_vector_float

sink_sptr.**pc_noutput_items_avg**(*sink_sptr self*) → float

sink_sptr.**pc_nproduced_avg**(*sink_sptr self*) → float

sink_sptr.**pc_output_buffers_full_avg**(*sink_sptr self*, *int which*) → float
pc_output_buffers_full_avg(sink_sptr self) -> pmt_vector_float

sink_sptr.**pc_throughput_avg**(*sink_sptr self*) → float

sink_sptr.**pc_work_time_avg**(*sink_sptr self*) → float

sink_sptr.**pc_work_time_total**(*sink_sptr self*) → float

sink_sptr.**sample_delay**(*sink_sptr self*, *int which*) → unsigned int

sink_sptr.**set_min_noutput_items**(*sink_sptr self*, *int m*)

sink_sptr.**set_thread_priority**(*sink_sptr self*, *int priority*) → int

sink_sptr.**thread_priority**(*sink_sptr self*) → int