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

gnuradio.wxgui

Provides a GUI interface using the Wx backend.

gnuradio.wxgui.**histo_sink_f**(*msg_queue_sptr msgq*) → histo_sink_f_sptr

Histogram module.

Constructor Specific Documentation:

Parameters: **msgq** –

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

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

declare_sample_delay(histo_sink_f_sptr self, unsigned int delay)

histo_sink_f_sptr.**get_frame_size**(*histo_sink_f_sptr self*) → unsigned int

histo_sink_f_sptr.**get_num_bins**(*histo_sink_f_sptr self*) → unsigned int

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

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

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

pc_input_buffers_full_avg(histo_sink_f_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(histo_sink_f_sptr self) -> pmt_vector_float

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

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

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

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

histo_sink_f_sptr.**set_frame_size**(*histo_sink_f_sptr self*, *unsigned int frame_size*)

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

histo_sink_f_sptr.**set_num_bins**(*histo_sink_f_sptr self*, *unsigned int num_bins*)

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

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

gnuradio.wxgui.**oscope_sink_f**(*double sampling_rate*, *msg_queue_sptr msgq*) → oscope_sink_f_sptr

Building block for python oscilloscope module.

Accepts multiple float streams.

Constructor Specific Documentation:

Parameters:
- **sampling_rate** –
- **msgq** –

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

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

declare_sample_delay(oscope_sink_f_sptr self, unsigned int delay)

oscope_sink_f_sptr.**get_decimation_count**(*oscope_sink_f_sptr self*) → int

oscope_sink_f_sptr.**get_samples_per_output_record**(*oscope_sink_f_sptr self*) → int

oscope_sink_f_sptr.**get_trigger_channel**(*oscope_sink_f_sptr self*) → int

oscope_sink_f_sptr.**get_trigger_level**(*oscope_sink_f_sptr self*) → double

oscope_sink_f_sptr.**get_trigger_mode**(*oscope_sink_f_sptr self*) → gr::wxgui::trigger_mode

oscope_sink_f_sptr.**get_trigger_slope**(*oscope_sink_f_sptr self*) → gr::wxgui::trigger_slope

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

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

oscope_sink_f_sptr.**num_channels**(*oscope_sink_f_sptr self*) → int

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

pc_input_buffers_full_avg(oscope_sink_f_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(oscope_sink_f_sptr self) -> pmt_vector_float

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

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

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

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

oscope_sink_f_sptr.**sample_rate**(*oscope_sink_f_sptr self*) → double

oscope_sink_f_sptr.**set_decimation_count**(*oscope_sink_f_sptr self*, *int decimation_count*) → bool

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

oscope_sink_f_sptr.**set_num_channels**(*oscope_sink_f_sptr self*, *int nchannels*) → bool

oscope_sink_f_sptr.**set_sample_rate**(*oscope_sink_f_sptr self*, *double sample_rate*) → bool

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

oscope_sink_f_sptr.**set_trigger_channel**(*oscope_sink_f_sptr self*, *int channel*) → bool

oscope_sink_f_sptr.**set_trigger_level**(*oscope_sink_f_sptr self*, *double trigger_level*) → bool

oscope_sink_f_sptr.**set_trigger_level_auto**(*oscope_sink_f_sptr self*) → bool

oscope_sink_f_sptr.**set_trigger_mode**(*oscope_sink_f_sptr self*, *gr::wxgui::trigger_mode mode*) → bool

oscope_sink_f_sptr.**set_trigger_slope**(*oscope_sink_f_sptr self*, *gr::wxgui::trigger_slope slope*) → bool

oscope_sink_f_sptr.**set_update_rate**(*oscope_sink_f_sptr self*, *double update_rate*) → bool

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

oscope_sink_f_sptr.**update_rate**(*oscope_sink_f_sptr self*) → double

oscope_sink_f_sptr.**work**(*oscope_sink_f_sptr self*, *int noutput_items*, *gr_vector_const_void_star & input_items*, *gr_vector_void_star & output_items*) → int