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

gnuradio.dtv

Blocks and utilities for digital TV module.

gnuradio.dtv.**atsc_deinterleaver**() → atsc_deinterleaver_sptr

ATSC deinterleave RS encoded ATSC data (atsc_mpeg_packet_rs_encoded > atsc_mpeg_packet_rs_encoded)

input: atsc_mpeg_packet_rs_encoded; output: atsc_mpeg_packet_rs_encoded.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_deinterleaver.

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

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

declare_sample_delay(atsc_deinterleaver_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_deinterleaver_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_deinterleaver_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_depad**() → atsc_depad_sptr

ATSC depad mpeg ts packets from 256 byte atsc_mpeg_packet to 188 byte char

input: atsc_mpeg_packet; output: unsigned char.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_depad.

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

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

declare_sample_delay(atsc_depad_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_depad_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_depad_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_derandomizer**() → atsc_derandomizer_sptr

ATSC “dewhiten” incoming mpeg transport stream packets

input: atsc_mpeg_packet_no_sync; output: atsc_mpeg_packet;.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_derandomizer.

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

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

declare_sample_delay(atsc_derandomizer_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_derandomizer_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_derandomizer_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_equalizer**() → atsc_equalizer_sptr

ATSC Receiver Equalizer.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_equalizer.

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

atsc_equalizer_sptr.**data**(*atsc_equalizer_sptr self*) → pmt_vector_float

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

declare_sample_delay(atsc_equalizer_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_equalizer_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_equalizer_sptr self) -> pmt_vector_float

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

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

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

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

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

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

atsc_equalizer_sptr.**taps**(*atsc_equalizer_sptr self*) → pmt_vector_float

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

gnuradio.dtv.**atsc_field_sync_mux**() → atsc_field_sync_mux_sptr

<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_field_sync_mux.

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

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

declare_sample_delay(atsc_field_sync_mux_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_field_sync_mux_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_field_sync_mux_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_fpll**(*float rate*) → atsc_fpll_sptr

ATSC Receiver FPLL.

This block is takes in a complex I/Q baseband stream from the receive filter and outputs the 8-level symbol

stream.

It does this by first locally generating a pilot tone and complex mixing with the input signal. This results in the pilot tone shifting to DC and places the signal in the upper sideband.

As no information is encoded in the phase of the waveform, the Q channel is then discarded, producing a real signal with the lower sideband restored.

The 8-level symbol stream still has a DC offset, and still requires symbol timing recovery.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_fpll.

param rate Sample rate of incoming stream

Parameters: **rate** –

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_fs_checker**() → atsc_fs_checker_sptr
ATSC Receiver FS_CHECKER.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_fs_checker.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_interleaver**() → atsc_interleaver_sptr

<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_interleaver.

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

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

declare_sample_delay(atsc_interleaver_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_interleaver_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_interleaver_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_pad**() → atsc_pad_sptr

<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_pad.

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

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

declare_sample_delay(atsc_pad_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_pad_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_randomizer**() → atsc_randomizer_sptr
<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_randomizer.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_rs_decoder**() → atsc_rs_decoder_sptr
ATSC Receiver Reed-Solomon Decoder.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_rs_decoder.

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

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

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

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

atsc_rs_decoder_sptr.**num_bad_packets**(*atsc_rs_decoder_sptr self*) → int
Returns the number of bad packets rejected by the decoder.

atsc_rs_decoder_sptr.**num_errors_corrected**(*atsc_rs_decoder_sptr self*) → int

Returns the number of errors corrected by the decoder.

atsc_rs_decoder_sptr.**num_packets**(*atsc_rs_decoder_sptr self*) → int

Returns the total number of packets seen by the decoder.

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

pc_input_buffers_full_avg(atsc_rs_decoder_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_rs_decoder_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_rs_encoder**() → atsc_rs_encoder_sptr

<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_rs_encoder.

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

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

declare_sample_delay(atsc_rs_encoder_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_rs_encoder_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_rs_encoder_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_sync**(*float rate*) → atsc_sync_sptr

ATSC Receiver SYNC.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_sync.

param rate Sample rate of incoming stream

Parameters: **rate** –

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

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

declare_sample_delay(atsc_sync_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_sync_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_sync_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_trellis_encoder**() → atsc_trellis_encoder_sptr

<+description of block+>

Constructor Specific Documentation:

Return a shared_ptr to a new instance of dtv::atsc_trellis_encoder.

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

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

declare_sample_delay(atsc_trellis_encoder_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(atsc_trellis_encoder_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_trellis_encoder_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**atsc_viterbi_decoder**() → atsc_viterbi_decoder_sptr

ATSC Viterbi Decoder.

Constructor Specific Documentation:

Make a new instance of gr::dtv::atsc_viterbi_decoder.

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

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

declare_sample_delay(atsc_viterbi_decoder_sptr self, unsigned int delay)

atsc_viterbi_decoder_sptr.**decoder_metrics**(*atsc_viterbi_decoder_sptr self*) → pmt_vector_float

For each decoder, returns the current best state of the decoding metrics.

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

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

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

pc_input_buffers_full_avg(atsc_viterbi_decoder_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(atsc_viterbi_decoder_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvb_bbheader_bb**(*gr::dtv::dvb_standard_t standard*, *gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvbs2_rolloff_factor_t rolloff*, *gr::dtv::dvbt2_inputmode_t mode*, *gr::dtv::dvbt2_inband_t inband*, *int fecblocks*, *int tsrate*) → dvb_bbheader_bb_sptr

Formats MPEG-2 Transport Stream packets into FEC baseband frames and adds a 10-byte header.

Input: 188-byte MPEG-2 Transport Stream packets.Output: Variable length FEC baseband frames (BBFRAME). The output frame length is based on the FEC rate.

Constructor Specific Documentation:

Create a baseband header formatter.

Parameters:

- **standard** – DVB standard (DVB-S2 or DVB-T2).
- **framesize** – FEC frame size (normal, medium or short).
- **rate** – FEC code rate.
- **rolloff** – DVB-S2 root-raised-cosine filter roll-off.
- **mode** – DVB-T2 input processing mode.
- **inband** – DVB-T2 Type B in-band signalling.
- **fecblocks** – DVB-T2 number of FEC block for in-band signalling.
- **tsrate** – DVB-T2 Transport Stream rate for in-band signalling.

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

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

declare_sample_delay(dvb_bbheader_bb_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvb_bbheader_bb_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvb_bbheader_bb_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvb_bbscrambler_bb**(*gr::dtv::dvb_standard_t standard*, *gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*) → dvb_bbscrambler_bb_sptr

Scrambles FEC baseband frames with a PRBS encoder.

Input: Variable length FEC baseband frames (BBFRAME)Output: Scrambled variable length FEC baseband frames (BBFRAME).

Constructor Specific Documentation:

Create a baseband frame scrambler.

Parameters:
- **standard** – DVB standard (DVB-S2 or DVB-T2).
- **framesize** – FEC frame size (normal, medium or short).
- **rate** – FEC code rate.

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

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

declare_sample_delay(dvb_bbscrambler_bb_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvb_bbscrambler_bb_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvb_bbscrambler_bb_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvb_bch_bb**(*gr::dtv::dvb_standard_t standard*, *gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*) → dvb_bch_bb_sptr

Encodes a BCH ((Bose, Chaudhuri, Hocquenghem) FEC.

Input: Variable length FEC baseband frames (BBFRAME)Output: Variable length FEC baseband frames with appended BCH (BCHFEC).

Constructor Specific Documentation:

Create a baseband frame BCH encoder.

Parameters:
- **standard** – DVB standard (DVB-S2 or DVB-T2).
- **framesize** – FEC frame size (normal, medium or short).
- **rate** – FEC code rate.

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

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

declare_sample_delay(dvb_bch_bb_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvb_bch_bb_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvb_bch_bb_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvb_ldpc_bb**(*gr::dtv::dvb_standard_t standard*, *gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvb_constellation_t constellation*) → dvb_ldpc_bb_sptr

Encodes a LDPC (Low-Density Parity-Check) FEC.

Input: Variable length FEC baseband frames with appended BCH (BCHFEC)Output: Normal, medium or short FEC baseband frames with appended LPDC (LDPCFEC).

Constructor Specific Documentation:

Create a baseband frame LDPC encoder.

Parameters:
- **standard** – DVB standard (DVB-S2 or DVB-T2).
- **framesize** – FEC frame size (normal, medium or short).
- **rate** – FEC code rate.
- **constellation** – DVB-S2 constellation.

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

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

declare_sample_delay(dvb_ldpc_bb_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvb_ldpc_bb_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvb_ldpc_bb_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbs2_interleaver_bb**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvb_constellation_t constellation*) → dvbs2_interleaver_bb_sptr

Bit interleaves DVB-S2 FEC baseband frames.

Input: Normal or short FEC baseband frames with appended LPDC (LDPCFEC)Output: Bit interleaved baseband frames.

Constructor Specific Documentation:

Create a DVB-S2 bit interleaver.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **rate** – FEC code rate.
- **constellation** – DVB-S2 constellation.

dvbs2_interleaver_bb_sptr.**active_thread_priority**(*dvbs2_interleaver_bb_sptr self*) → int

dvbs2_interleaver_bb_sptr.**declare_sample_delay**(*dvbs2_interleaver_bb_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbs2_interleaver_bb_sptr self, unsigned int delay)

dvbs2_interleaver_bb_sptr.**message_subscribers**(*dvbs2_interleaver_bb_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbs2_interleaver_bb_sptr.**min_noutput_items**(*dvbs2_interleaver_bb_sptr self*) → int

dvbs2_interleaver_bb_sptr.**pc_input_buffers_full_avg**(*dvbs2_interleaver_bb_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbs2_interleaver_bb_sptr self) -> pmt_vector_float

dvbs2_interleaver_bb_sptr.**pc_noutput_items_avg**(*dvbs2_interleaver_bb_sptr self*) → float

dvbs2_interleaver_bb_sptr.**pc_nproduced_avg**(*dvbs2_interleaver_bb_sptr self*) → float

dvbs2_interleaver_bb_sptr.**pc_output_buffers_full_avg**(*dvbs2_interleaver_bb_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbs2_interleaver_bb_sptr self) -> pmt_vector_float

dvbs2_interleaver_bb_sptr.**pc_throughput_avg**(*dvbs2_interleaver_bb_sptr self*) → float

dvbs2_interleaver_bb_sptr.**pc_work_time_avg**(*dvbs2_interleaver_bb_sptr self*) → float

dvbs2_interleaver_bb_sptr.**pc_work_time_total**(*dvbs2_interleaver_bb_sptr self*) → float

dvbs2_interleaver_bb_sptr.**sample_delay**(*dvbs2_interleaver_bb_sptr self*, *int which*) → unsigned int

dvbs2_interleaver_bb_sptr.**set_min_noutput_items**(*dvbs2_interleaver_bb_sptr self*, *int m*)

dvbs2_interleaver_bb_sptr.**set_thread_priority**(*dvbs2_interleaver_bb_sptr self*, *int priority*) → int

dvbs2_interleaver_bb_sptr.**thread_priority**(*dvbs2_interleaver_bb_sptr self*) → int

gnuradio.dtv.**dvbs2_modulator_bc**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*,

gr::dtv::dvb_constellation_t constellation, *gr::dtv::dvbs2_interpolation_t interpolation*) → dvbs2_modulator_bc_sptr

Modulates DVB-S2 frames.

Input: Bit interleaved baseband frames. Output: QPSK, 8PSK, 16APSK or 32APSK modulated complex IQ values (XFECFRAME).

Constructor Specific Documentation:

Create a DVB-S2 constellation modulator.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **rate** – FEC code rate.
- **constellation** – DVB-S2 constellation.
- **interpolation** – 2X zero stuffing interpolation (on/off).

dvbs2_modulator_bc_sptr.**active_thread_priority**(*dvbs2_modulator_bc_sptr self*) → int

dvbs2_modulator_bc_sptr.**declare_sample_delay**(*dvbs2_modulator_bc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbs2_modulator_bc_sptr self, unsigned int delay)

dvbs2_modulator_bc_sptr.**message_subscribers**(*dvbs2_modulator_bc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbs2_modulator_bc_sptr.**min_noutput_items**(*dvbs2_modulator_bc_sptr self*) → int

dvbs2_modulator_bc_sptr.**pc_input_buffers_full_avg**(*dvbs2_modulator_bc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbs2_modulator_bc_sptr self) -> pmt_vector_float

dvbs2_modulator_bc_sptr.**pc_noutput_items_avg**(*dvbs2_modulator_bc_sptr self*) → float

dvbs2_modulator_bc_sptr.**pc_nproduced_avg**(*dvbs2_modulator_bc_sptr self*) → float

dvbs2_modulator_bc_sptr.**pc_output_buffers_full_avg**(*dvbs2_modulator_bc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbs2_modulator_bc_sptr self) -> pmt_vector_float

dvbs2_modulator_bc_sptr.**pc_throughput_avg**(*dvbs2_modulator_bc_sptr self*) → float

dvbs2_modulator_bc_sptr.**pc_work_time_avg**(*dvbs2_modulator_bc_sptr self*) → float

dvbs2_modulator_bc_sptr.**pc_work_time_total**(*dvbs2_modulator_bc_sptr self*) → float

dvbs2_modulator_bc_sptr.**sample_delay**(*dvbs2_modulator_bc_sptr self*, *int which*) → unsigned int

dvbs2_modulator_bc_sptr.**set_min_noutput_items**(*dvbs2_modulator_bc_sptr self*, *int m*)

dvbs2_modulator_bc_sptr.**set_thread_priority**(*dvbs2_modulator_bc_sptr self*, *int priority*) → int

dvbs2_modulator_bc_sptr.**thread_priority**(*dvbs2_modulator_bc_sptr self*) → int

gnuradio.dtv.**dvbs2_physical_cc**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbs2_pilots_t pilots*, *int goldcode*) → dvbs2_physical_cc_sptr

Signals DVB-S2 physical layer frames.

Input: QPSK, 8PSK, 16APSK or 32APSK modulated complex IQ values (XFECFRAME). Output: DVB-S2 PLFRAME.

Constructor Specific Documentation:

Create a DVB-S2 physical layer framer.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **rate** – FEC code rate.
- **constellation** – DVB-S2 constellation.
- **pilots** – pilot symbols (on/off).
- **goldcode** – physical layer scrambler Gold code (0 to 262141 inclusive).

dvbs2_physical_cc_sptr.**active_thread_priority**(*dvbs2_physical_cc_sptr self*) → int

dvbs2_physical_cc_sptr.**declare_sample_delay**(*dvbs2_physical_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbs2_physical_cc_sptr self, unsigned int delay)

dvbs2_physical_cc_sptr.**message_subscribers**(*dvbs2_physical_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbs2_physical_cc_sptr.**min_noutput_items**(*dvbs2_physical_cc_sptr self*) → int

dvbs2_physical_cc_sptr.**pc_input_buffers_full_avg**(*dvbs2_physical_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbs2_physical_cc_sptr self) -> pmt_vector_float

dvbs2_physical_cc_sptr.**pc_noutput_items_avg**(*dvbs2_physical_cc_sptr self*) → float

dvbs2_physical_cc_sptr.**pc_nproduced_avg**(*dvbs2_physical_cc_sptr self*) → float

dvbs2_physical_cc_sptr.**pc_output_buffers_full_avg**(*dvbs2_physical_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbs2_physical_cc_sptr self) -> pmt_vector_float

dvbs2_physical_cc_sptr.**pc_throughput_avg**(*dvbs2_physical_cc_sptr self*) → float

dvbs2_physical_cc_sptr.**pc_work_time_avg**(*dvbs2_physical_cc_sptr self*) → float

dvbs2_physical_cc_sptr.**pc_work_time_total**(*dvbs2_physical_cc_sptr self*) → float

dvbs2_physical_cc_sptr.**sample_delay**(*dvbs2_physical_cc_sptr self*, *int which*) → unsigned int

dvbs2_physical_cc_sptr.**set_min_noutput_items**(*dvbs2_physical_cc_sptr self*, *int m*)

dvbs2_physical_cc_sptr.**set_thread_priority**(*dvbs2_physical_cc_sptr self*, *int priority*) → int

dvbs2_physical_cc_sptr.**thread_priority**(*dvbs2_physical_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_cellinterleaver_cc**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_constellation_t constellation*, *int fecblocks*, *int tiblocks*) → dvbt2_cellinterleaver_cc_sptr

Cell and time interleaves QPSK/QAM modulated cells.

Input: QPSK, 16QAM, 64QAM or 256QAM modulated cellsOutput: Cell and time interleaved QPSK, 16QAM, 64QAM or 256QAM modulated cells.

Constructor Specific Documentation:

Create a DVB-T2 cell and time interleaver.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **constellation** – DVB-T2 constellation.
- **fecblocks** – number of FEC frames in a T2 frame.
- **tiblocks** – number of time interleaving blocks in a T2 frame.

dvbt2_cellinterleaver_cc_sptr.**active_thread_priority**(*dvbt2_cellinterleaver_cc_sptr self*) → int

dvbt2_cellinterleaver_cc_sptr.**declare_sample_delay**(*dvbt2_cellinterleaver_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_cellinterleaver_cc_sptr self, unsigned int delay)

dvbt2_cellinterleaver_cc_sptr.**message_subscribers**(*dvbt2_cellinterleaver_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_cellinterleaver_cc_sptr.**min_noutput_items**(*dvbt2_cellinterleaver_cc_sptr self*) → int

dvbt2_cellinterleaver_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_cellinterleaver_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_cellinterleaver_cc_sptr self) -> pmt_vector_float

dvbt2_cellinterleaver_cc_sptr.**pc_noutput_items_avg**(*dvbt2_cellinterleaver_cc_sptr self*) → float

dvbt2_cellinterleaver_cc_sptr.**pc_nproduced_avg**(*dvbt2_cellinterleaver_cc_sptr self*) → float

dvbt2_cellinterleaver_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_cellinterleaver_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_cellinterleaver_cc_sptr self) -> pmt_vector_float

dvbt2_cellinterleaver_cc_sptr.**pc_throughput_avg**(*dvbt2_cellinterleaver_cc_sptr self*) → float

dvbt2_cellinterleaver_cc_sptr.**pc_work_time_avg**(*dvbt2_cellinterleaver_cc_sptr self*) → float

dvbt2_cellinterleaver_cc_sptr.**pc_work_time_total**(*dvbt2_cellinterleaver_cc_sptr self*) → float

dvbt2_cellinterleaver_cc_sptr.**sample_delay**(*dvbt2_cellinterleaver_cc_sptr self*, *int which*) → unsigned int

dvbt2_cellinterleaver_cc_sptr.**set_min_noutput_items**(*dvbt2_cellinterleaver_cc_sptr self*, *int m*)

dvbt2_cellinterleaver_cc_sptr.**set_thread_priority**(*dvbt2_cellinterleaver_cc_sptr self*, *int priority*) → int

dvbt2_cellinterleaver_cc_sptr.**thread_priority**(*dvbt2_cellinterleaver_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_framemapper_cc**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt2_rotation_t rotation*, *int fecblocks*, *int tiblocks*, *gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvb_guardinterval_t guardinterval*, *gr::dtv::dvbt2_l1constellation_t l1constellation*, *gr::dtv::dvbt2_pilotpattern_t pilotpattern*, *int t2frames*, *int numdatasyms*, *gr::dtv::dvbt2_papr_t paprmode*, *gr::dtv::dvbt2_version_t version*, *gr::dtv::dvbt2_preamble_t preamble*, *gr::dtv::dvbt2_inputmode_t inputmode*, *gr::dtv::dvbt2_reservedbiasbits_t reservedbiasbits*, *gr::dtv::dvbt2_l1scrambled_t l1scrambled*, *gr::dtv::dvbt2_inband_t inband*) → dvbt2_framemapper_cc_sptr

Maps T2 frames.

Input: Cell and time interleaved QPSK, 16QAM, 64QAM or 256QAM modulated cells. Output: T2 frame.

Constructor Specific Documentation:

Create a DVB-T2 frame mapper.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **rate** – FEC code rate.
- **constellation** – DVB-T2 constellation.
- **rotation** – DVB-T2 constellation rotation (on or off).
- **fecblocks** – number of FEC frames in a T2 frame.
- **tiblocks** – number of time interleaving blocks in a T2 frame.
- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **guardinterval** – OFDM ISI guard interval.
- **l1constellation** – L1 constellation.
- **pilotpattern** – DVB-T2 pilot pattern (PP1 - PP8).
- **t2frames** – number of T2 frames in a super-frame.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **paprmode** – PAPR reduction mode.
- **version** – DVB-T2 specification version.
- **preamble** – P1 symbol preamble format.
- **inputmode** – Baseband Header mode.
- **reservedbiasbits** – set all L1 bias bits to 1 (on or off).
- **l1scrambled** – scramble L1 post signalling (on or off).
- **inband** – In-band type B signalling (on or off).

dvbt2_framemapper_cc_sptr.**active_thread_priority**(*dvbt2_framemapper_cc_sptr self*) → int

dvbt2_framemapper_cc_sptr.**declare_sample_delay**(*dvbt2_framemapper_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_framemapper_cc_sptr self, unsigned int delay)

dvbt2_framemapper_cc_sptr.**message_subscribers**(*dvbt2_framemapper_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_framemapper_cc_sptr.**min_noutput_items**(*dvbt2_framemapper_cc_sptr self*) → int

dvbt2_framemapper_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_framemapper_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_framemapper_cc_sptr self) -> pmt_vector_float

dvbt2_framemapper_cc_sptr.**pc_noutput_items_avg**(*dvbt2_framemapper_cc_sptr self*) → float

dvbt2_framemapper_cc_sptr.**pc_nproduced_avg**(*dvbt2_framemapper_cc_sptr self*) → float

dvbt2_framemapper_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_framemapper_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_framemapper_cc_sptr self) -> pmt_vector_float

dvbt2_framemapper_cc_sptr.**pc_throughput_avg**(*dvbt2_framemapper_cc_sptr self*) → float

dvbt2_framemapper_cc_sptr.**pc_work_time_avg**(*dvbt2_framemapper_cc_sptr self*) → float

dvbt2_framemapper_cc_sptr.**pc_work_time_total**(*dvbt2_framemapper_cc_sptr self*) → float

dvbt2_framemapper_cc_sptr.**sample_delay**(*dvbt2_framemapper_cc_sptr self*, *int which*) → unsigned int

dvbt2_framemapper_cc_sptr.**set_min_noutput_items**(*dvbt2_framemapper_cc_sptr self*, *int m*)

dvbt2_framemapper_cc_sptr.**set_thread_priority**(*dvbt2_framemapper_cc_sptr self*, *int priority*) → int

dvbt2_framemapper_cc_sptr.**thread_priority**(*dvbt2_framemapper_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_freqinterleaver_cc**(*gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvbt2_pilotpattern_t pilotpattern*, *gr::dtv::dvb_guardinterval_t guardinterval*, *int numdatasyms*, *gr::dtv::dvbt2_papr_t paprmode*, *gr::dtv::dvbt2_version_t version*, *gr::dtv::dvbt2_preamble_t preamble*) → dvbt2_freqinterleaver_cc_sptr

Frequency interleaves a T2 frame.

Input: T2 frame. Output: Frequency interleaved T2 frame.

Constructor Specific Documentation:

Create a DVB-T2 frequency interleaver.

Parameters:
- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **pilotpattern** – DVB-T2 pilot pattern (PP1 - PP8).
- **guardinterval** – OFDM ISI guard interval.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **paprmode** – PAPR reduction mode.
- **version** – DVB-T2 specification version.
- **preamble** – P1 symbol preamble format.

dvbt2_freqinterleaver_cc_sptr.**active_thread_priority**(*dvbt2_freqinterleaver_cc_sptr self*) → int

dvbt2_freqinterleaver_cc_sptr.**declare_sample_delay**(*dvbt2_freqinterleaver_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_freqinterleaver_cc_sptr self, unsigned int delay)

dvbt2_freqinterleaver_cc_sptr.**message_subscribers**(*dvbt2_freqinterleaver_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_freqinterleaver_cc_sptr.**min_noutput_items**(*dvbt2_freqinterleaver_cc_sptr self*) → int

dvbt2_freqinterleaver_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_freqinterleaver_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_freqinterleaver_cc_sptr self) -> pmt_vector_float

dvbt2_freqinterleaver_cc_sptr.**pc_noutput_items_avg**(*dvbt2_freqinterleaver_cc_sptr self*) → float

dvbt2_freqinterleaver_cc_sptr.**pc_nproduced_avg**(*dvbt2_freqinterleaver_cc_sptr self*) → float

dvbt2_freqinterleaver_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_freqinterleaver_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_freqinterleaver_cc_sptr self) -> pmt_vector_float

dvbt2_freqinterleaver_cc_sptr.**pc_throughput_avg**(*dvbt2_freqinterleaver_cc_sptr self*) → float

dvbt2_freqinterleaver_cc_sptr.**pc_work_time_avg**(*dvbt2_freqinterleaver_cc_sptr self*) → float

dvbt2_freqinterleaver_cc_sptr.**pc_work_time_total**(*dvbt2_freqinterleaver_cc_sptr self*) → float

dvbt2_freqinterleaver_cc_sptr.**sample_delay**(*dvbt2_freqinterleaver_cc_sptr self*, *int which*) → unsigned int

dvbt2_freqinterleaver_cc_sptr.**set_min_noutput_items**(*dvbt2_freqinterleaver_cc_sptr self*, *int m*)

dvbt2_freqinterleaver_cc_sptr.**set_thread_priority**(*dvbt2_freqinterleaver_cc_sptr self*, *int priority*) → int

dvbt2_freqinterleaver_cc_sptr.**thread_priority**(*dvbt2_freqinterleaver_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_interleaver_bb**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_code_rate_t rate*, *gr::dtv::dvb_constellation_t constellation*) → dvbt2_interleaver_bb_sptr

Bit interleaves DVB-T2 FEC baseband frames.

Input: Normal or short FEC baseband frames with appended LPDC (LDPCFEC)Output: Bit interleaved (with column twist and bit to cell word de-multiplexed) cells.

Constructor Specific Documentation:

Create a DVB-T2 bit interleaver.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **rate** – FEC code rate.
- **constellation** – DVB-T2 constellation.

dvbt2_interleaver_bb_sptr.**active_thread_priority**(*dvbt2_interleaver_bb_sptr self*) → int

dvbt2_interleaver_bb_sptr.**declare_sample_delay**(*dvbt2_interleaver_bb_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_interleaver_bb_sptr self, unsigned int delay)

dvbt2_interleaver_bb_sptr.**message_subscribers**(*dvbt2_interleaver_bb_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_interleaver_bb_sptr.**min_noutput_items**(*dvbt2_interleaver_bb_sptr self*) → int

dvbt2_interleaver_bb_sptr.**pc_input_buffers_full_avg**(*dvbt2_interleaver_bb_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_interleaver_bb_sptr self) -> pmt_vector_float

dvbt2_interleaver_bb_sptr.**pc_noutput_items_avg**(*dvbt2_interleaver_bb_sptr self*) → float

dvbt2_interleaver_bb_sptr.**pc_nproduced_avg**(*dvbt2_interleaver_bb_sptr self*) → float

dvbt2_interleaver_bb_sptr.**pc_output_buffers_full_avg**(*dvbt2_interleaver_bb_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_interleaver_bb_sptr self) -> pmt_vector_float

dvbt2_interleaver_bb_sptr.**pc_throughput_avg**(*dvbt2_interleaver_bb_sptr self*) → float

dvbt2_interleaver_bb_sptr.**pc_work_time_avg**(*dvbt2_interleaver_bb_sptr self*) → float

dvbt2_interleaver_bb_sptr.**pc_work_time_total**(*dvbt2_interleaver_bb_sptr self*) → float

dvbt2_interleaver_bb_sptr.**sample_delay**(*dvbt2_interleaver_bb_sptr self*, *int which*) → unsigned int

dvbt2_interleaver_bb_sptr.**set_min_noutput_items**(*dvbt2_interleaver_bb_sptr self*, *int m*)

dvbt2_interleaver_bb_sptr.**set_thread_priority**(*dvbt2_interleaver_bb_sptr self*, *int priority*) → int

dvbt2_interleaver_bb_sptr.**thread_priority**(*dvbt2_interleaver_bb_sptr self*) → int

gnuradio.dtv.**dvbt2_miso_cc**(*gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvbt2_pilotpattern_t pilotpattern*, *gr::dtv::dvb_guardinterval_t guardinterval*, *int numdatasyms*, *gr::dtv::dvbt2_papr_t paprmode*) → dvbt2_miso_cc_sptr

Splits the stream for MISO (Multiple Input Single Output).

Input: Frequency interleaved T2 frame.Output1: Frequency interleaved T2 frame (copy of input).Output2: Frequency interleaved T2 frame with modified Alamouti processing.

Constructor Specific Documentation:

Create a MISO processor.

Parameters:
- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **pilotpattern** – DVB-T2 pilot pattern (PP1 - PP8).
- **guardinterval** – OFDM ISI guard interval.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **paprmode** – PAPR reduction mode.

dvbt2_miso_cc_sptr.**active_thread_priority**(*dvbt2_miso_cc_sptr self*) → int

dvbt2_miso_cc_sptr.**declare_sample_delay**(*dvbt2_miso_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_miso_cc_sptr self, unsigned int delay)

dvbt2_miso_cc_sptr.**message_subscribers**(*dvbt2_miso_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_miso_cc_sptr.**min_noutput_items**(*dvbt2_miso_cc_sptr self*) → int

dvbt2_miso_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_miso_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_miso_cc_sptr self) -> pmt_vector_float

dvbt2_miso_cc_sptr.**pc_noutput_items_avg**(*dvbt2_miso_cc_sptr self*) → float

dvbt2_miso_cc_sptr.**pc_nproduced_avg**(*dvbt2_miso_cc_sptr self*) → float

dvbt2_miso_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_miso_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_miso_cc_sptr self) -> pmt_vector_float

dvbt2_miso_cc_sptr.**pc_throughput_avg**(*dvbt2_miso_cc_sptr self*) → float

dvbt2_miso_cc_sptr.**pc_work_time_avg**(*dvbt2_miso_cc_sptr self*) → float

dvbt2_miso_cc_sptr.**pc_work_time_total**(*dvbt2_miso_cc_sptr self*) → float

dvbt2_miso_cc_sptr.**sample_delay**(*dvbt2_miso_cc_sptr self*, *int which*) → unsigned int

dvbt2_miso_cc_sptr.**set_min_noutput_items**(*dvbt2_miso_cc_sptr self*, *int m*)

dvbt2_miso_cc_sptr.**set_thread_priority**(*dvbt2_miso_cc_sptr self*, *int priority*) → int

dvbt2_miso_cc_sptr.**thread_priority**(*dvbt2_miso_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_modulator_bc**(*gr::dtv::dvb_framesize_t framesize*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt2_rotation_t rotation*) → dvbt2_modulator_bc_sptr

Modulates DVB-T2 cells.

Input: Bit interleaved (with column twist and bit to cell word de-multiplexing) cells. Output: QPSK, 16QAM, 64QAM or 256QAM modulated complex IQ values (cells).

Constructor Specific Documentation:

Create a DVB-T2 constellation modulator.

Parameters:
- **framesize** – FEC frame size (normal or short).
- **constellation** – DVB-T2 constellation.
- **rotation** – DVB-T2 constellation rotation (on or off).

dvbt2_modulator_bc_sptr.**active_thread_priority**(*dvbt2_modulator_bc_sptr self*) → int

dvbt2_modulator_bc_sptr.**declare_sample_delay**(*dvbt2_modulator_bc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_modulator_bc_sptr self, unsigned int delay)

dvbt2_modulator_bc_sptr.**message_subscribers**(*dvbt2_modulator_bc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_modulator_bc_sptr.**min_noutput_items**(*dvbt2_modulator_bc_sptr self*) → int

dvbt2_modulator_bc_sptr.**pc_input_buffers_full_avg**(*dvbt2_modulator_bc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_modulator_bc_sptr self) -> pmt_vector_float

dvbt2_modulator_bc_sptr.**pc_noutput_items_avg**(*dvbt2_modulator_bc_sptr self*) → float

dvbt2_modulator_bc_sptr.**pc_nproduced_avg**(*dvbt2_modulator_bc_sptr self*) → float

dvbt2_modulator_bc_sptr.**pc_output_buffers_full_avg**(*dvbt2_modulator_bc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_modulator_bc_sptr self) -> pmt_vector_float

dvbt2_modulator_bc_sptr.**pc_throughput_avg**(*dvbt2_modulator_bc_sptr self*) → float

dvbt2_modulator_bc_sptr.**pc_work_time_avg**(*dvbt2_modulator_bc_sptr self*) → float

dvbt2_modulator_bc_sptr.**pc_work_time_total**(*dvbt2_modulator_bc_sptr self*) → float

dvbt2_modulator_bc_sptr.**sample_delay**(*dvbt2_modulator_bc_sptr self*, *int which*) → unsigned int

dvbt2_modulator_bc_sptr.**set_min_noutput_items**(*dvbt2_modulator_bc_sptr self*, *int m*)

dvbt2_modulator_bc_sptr.**set_thread_priority**(*dvbt2_modulator_bc_sptr self*, *int priority*) → int

dvbt2_modulator_bc_sptr.**thread_priority**(*dvbt2_modulator_bc_sptr self*) → int

gnuradio.dtv.**dvbt2_p1insertion_cc**(*gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvb_guardinterval_t guardinterval*, *int numdatasyms*, *gr::dtv::dvbt2_preamble_t preamble*, *gr::dtv::dvbt2_showlevels_t showlevels*, *float vclip*) → dvbt2_p1insertion_cc_sptr

Inserts a P1 symbol.

Input: OFDM T2 frame. Output: OFDM T2 frame with P1 symbol.

Constructor Specific Documentation:

Create a P1 symbol inserter.

Parameters:
- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **guardinterval** – OFDM ISI guard interval.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **preamble** – P1 symbol preamble format.
- **showlevels** – print peak IQ levels.
- **vclip** – set peak IQ level threshold.

dvbt2_p1insertion_cc_sptr.**active_thread_priority**(*dvbt2_p1insertion_cc_sptr self*) → int

dvbt2_p1insertion_cc_sptr.**declare_sample_delay**(*dvbt2_p1insertion_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_p1insertion_cc_sptr self, unsigned int delay)

dvbt2_p1insertion_cc_sptr.**message_subscribers**(*dvbt2_p1insertion_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_p1insertion_cc_sptr.**min_noutput_items**(*dvbt2_p1insertion_cc_sptr self*) → int

dvbt2_p1insertion_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_p1insertion_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_p1insertion_cc_sptr self) -> pmt_vector_float

dvbt2_p1insertion_cc_sptr.**pc_noutput_items_avg**(*dvbt2_p1insertion_cc_sptr self*) → float

dvbt2_p1insertion_cc_sptr.**pc_nproduced_avg**(*dvbt2_p1insertion_cc_sptr self*) → float

dvbt2_p1insertion_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_p1insertion_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_p1insertion_cc_sptr self) -> pmt_vector_float

dvbt2_p1insertion_cc_sptr.**pc_throughput_avg**(*dvbt2_p1insertion_cc_sptr self*) → float

dvbt2_p1insertion_cc_sptr.**pc_work_time_avg**(*dvbt2_p1insertion_cc_sptr self*) → float

dvbt2_p1insertion_cc_sptr.**pc_work_time_total**(*dvbt2_p1insertion_cc_sptr self*) → float

dvbt2_p1insertion_cc_sptr.**sample_delay**(*dvbt2_p1insertion_cc_sptr self*, *int which*) → unsigned int

dvbt2_p1insertion_cc_sptr.**set_min_noutput_items**(*dvbt2_p1insertion_cc_sptr self*, *int m*)

dvbt2_p1insertion_cc_sptr.**set_thread_priority**(*dvbt2_p1insertion_cc_sptr self*, *int priority*) → int

dvbt2_p1insertion_cc_sptr.**thread_priority**(*dvbt2_p1insertion_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_paprtr_cc**(*gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvbt2_pilotpattern_t pilotpattern*, *gr::dtv::dvb_guardinterval_t guardinterval*, *int numdatasyms*, *gr::dtv::dvbt2_papr_t paprmode*, *gr::dtv::dvbt2_version_t version*, *float vclip*, *int iterations*, *int vlength*) → dvbt2_paprtr_cc_sptr

Peak to Average Power Ratio (PAPR) reduction.

Input: A T2 frame of OFDM symbols. Output: A T2 frame of PAPR reduced OFDM symbols.

Constructor Specific Documentation:

Create a PAPR reducer.

Parameters:
- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **pilotpattern** – DVB-T2 pilot pattern (PP1 - PP8).
- **guardinterval** – OFDM ISI guard interval.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **paprmode** – PAPR reduction mode.
- **version** – DVB-T2 specification version.
- **vclip** – PAPR clipping level.
- **iterations** – PAPR algorithm number of iterations.
- **vlength** – input and output vector length.

dvbt2_paprtr_cc_sptr.**active_thread_priority**(*dvbt2_paprtr_cc_sptr self*) → int

dvbt2_paprtr_cc_sptr.**declare_sample_delay**(*dvbt2_paprtr_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_paprtr_cc_sptr self, unsigned int delay)

dvbt2_paprtr_cc_sptr.**message_subscribers**(*dvbt2_paprtr_cc_sptr self*, *swig_int_ptr which_port*) →

swig_int_ptr

dvbt2_paprtr_cc_sptr.**min_noutput_items**(*dvbt2_paprtr_cc_sptr self*) → int

dvbt2_paprtr_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_paprtr_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_paprtr_cc_sptr self) -> pmt_vector_float

dvbt2_paprtr_cc_sptr.**pc_noutput_items_avg**(*dvbt2_paprtr_cc_sptr self*) → float

dvbt2_paprtr_cc_sptr.**pc_nproduced_avg**(*dvbt2_paprtr_cc_sptr self*) → float

dvbt2_paprtr_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_paprtr_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_paprtr_cc_sptr self) -> pmt_vector_float

dvbt2_paprtr_cc_sptr.**pc_throughput_avg**(*dvbt2_paprtr_cc_sptr self*) → float

dvbt2_paprtr_cc_sptr.**pc_work_time_avg**(*dvbt2_paprtr_cc_sptr self*) → float

dvbt2_paprtr_cc_sptr.**pc_work_time_total**(*dvbt2_paprtr_cc_sptr self*) → float

dvbt2_paprtr_cc_sptr.**sample_delay**(*dvbt2_paprtr_cc_sptr self*, *int which*) → unsigned int

dvbt2_paprtr_cc_sptr.**set_min_noutput_items**(*dvbt2_paprtr_cc_sptr self*, *int m*)

dvbt2_paprtr_cc_sptr.**set_thread_priority**(*dvbt2_paprtr_cc_sptr self*, *int priority*) → int

dvbt2_paprtr_cc_sptr.**thread_priority**(*dvbt2_paprtr_cc_sptr self*) → int

gnuradio.dtv.**dvbt2_pilotgenerator_cc**(*gr::dtv::dvbt2_extended_carrier_t carriermode*, *gr::dtv::dvbt2_fftsize_t fftsize*, *gr::dtv::dvbt2_pilotpattern_t pilotpattern*, *gr::dtv::dvb_guardinterval_t guardinterval*, *int numdatasyms*, *gr::dtv::dvbt2_papr_t paprmode*, *gr::dtv::dvbt2_version_t version*, *gr::dtv::dvbt2_preamble_t preamble*, *gr::dtv::dvbt2_misogroup_t misogroup*, *gr::dtv::dvbt2_equalization_t equalization*, *gr::dtv::dvbt2_bandwidth_t bandwidth*, *int vlength*) → dvbt2_pilotgenerator_cc_sptr

Adds pilots to T2 frames.

Input: Frequency interleaved T2 frame. Output: T2 frame with pilots (in time domain).

Constructor Specific Documentation:

Create a DVB-T2 pilot generator.

Parameters:

- **carriermode** – number of carriers (normal or extended).
- **fftsize** – OFDM IFFT size.
- **pilotpattern** – DVB-T2 pilot pattern (PP1 - PP8).
- **guardinterval** – OFDM ISI guard interval.
- **numdatasyms** – number of OFDM symbols in a T2 frame.
- **paprmode** – PAPR reduction mode.
- **version** – DVB-T2 specification version.
- **preamble** – P1 symbol preamble format.
- **misogroup** – MISO transmitter ID.
- **equalization** – sin(x)/x DAC equalization (on or off).
- **bandwidth** – sin(x)/x equalization bandwidth.
- **vlength** – output vector length.

dvbt2_pilotgenerator_cc_sptr.**active_thread_priority**(*dvbt2_pilotgenerator_cc_sptr self*) → int

dvbt2_pilotgenerator_cc_sptr.**declare_sample_delay**(*dvbt2_pilotgenerator_cc_sptr self*, *int which*, *int delay*)

declare_sample_delay(dvbt2_pilotgenerator_cc_sptr self, unsigned int delay)

dvbt2_pilotgenerator_cc_sptr.**message_subscribers**(*dvbt2_pilotgenerator_cc_sptr self*, *swig_int_ptr which_port*) → swig_int_ptr

dvbt2_pilotgenerator_cc_sptr.**min_noutput_items**(*dvbt2_pilotgenerator_cc_sptr self*) → int

dvbt2_pilotgenerator_cc_sptr.**pc_input_buffers_full_avg**(*dvbt2_pilotgenerator_cc_sptr self*, *int which*) → float

pc_input_buffers_full_avg(dvbt2_pilotgenerator_cc_sptr self) -> pmt_vector_float

dvbt2_pilotgenerator_cc_sptr.**pc_noutput_items_avg**(*dvbt2_pilotgenerator_cc_sptr self*) → float

dvbt2_pilotgenerator_cc_sptr.**pc_nproduced_avg**(*dvbt2_pilotgenerator_cc_sptr self*) → float

dvbt2_pilotgenerator_cc_sptr.**pc_output_buffers_full_avg**(*dvbt2_pilotgenerator_cc_sptr self*, *int which*) → float

pc_output_buffers_full_avg(dvbt2_pilotgenerator_cc_sptr self) -> pmt_vector_float

dvbt2_pilotgenerator_cc_sptr.**pc_throughput_avg**(*dvbt2_pilotgenerator_cc_sptr self*) → float

dvbt2_pilotgenerator_cc_sptr.**pc_work_time_avg**(*dvbt2_pilotgenerator_cc_sptr self*) → float

dvbt2_pilotgenerator_cc_sptr.**pc_work_time_total**(*dvbt2_pilotgenerator_cc_sptr self*) → float

dvbt2_pilotgenerator_cc_sptr.**sample_delay**(*dvbt2_pilotgenerator_cc_sptr self*, *int which*) → unsigned int

dvbt2_pilotgenerator_cc_sptr.**set_min_noutput_items**(*dvbt2_pilotgenerator_cc_sptr self*, *int m*)

dvbt2_pilotgenerator_cc_sptr.**set_thread_priority**(*dvbt2_pilotgenerator_cc_sptr self*, *int priority*) → int

dvbt2_pilotgenerator_cc_sptr.**thread_priority**(*dvbt2_pilotgenerator_cc_sptr self*) → int

gnuradio.dtv.**dvbt_bit_inner_interleaver**(*int nsize*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt_hierarchy_t hierarchy*, *gr::dtv::dvbt_transmission_mode_t transmission*) → dvbt_bit_inner_interleaver_sptr

Bit Inner interleaver.

ETSI EN 300 744 Clause 4.3.4.1Data Input format: 000000X0X1 - QPSK.0000X0X1X2X3 - 16QAM. 00X0X1X2X3X4X5 - 64QAMData Output format: 000000B0B1 - QPSK.0000B0B1B2B3 - 16QAM. 00B0B1B2B3B4B5 - 64QAM. bit interleaver block size is 126.

Constructor Specific Documentation:

Create a Bit Inner interleaver.

Parameters:
- **nsize** – length of input stream.
- **constellation** – constellation used.
- **hierarchy** – hierarchy used.
- **transmission** – transmission mode used.

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

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

declare_sample_delay(dvbt_bit_inner_interleaver_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_bit_inner_interleaver_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_bit_inner_interleaver_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_convolutional_interleaver**(*int nsize*, *int I*, *int M*) → dvbt_convolutional_interleaver_sptr

Convolutional interleaver.

ETSI EN 300 744 Clause 4.3.1Forney (Ramsey type III) convolutional interleaver.Input: Blocks of I bytes size. Output: Stream of 1 byte elements.

Constructor Specific Documentation:

Create a DVB-T convolutional interleaver.

Parameters:
- **nsize** – number of blocks to process.
- **I** – size of a block.
- **M** – depth length for each element in shift registers.

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

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

declare_sample_delay(dvbt_convolutional_interleaver_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_convolutional_interleaver_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_convolutional_interleaver_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_energy_dispersal**(*int nsize*) → dvbt_energy_dispersal_sptr

Energy dispersal.

ETSI EN 300 744 - Clause 4.3.1Input - MPEG-2 transport packets (including sync - 0x47).Output - Randomized MPEG-2 transport packets.If first byte is not a SYNC then look for it.First sync in a row of 8 packets is reversed - 0xB8. Block size is 188 bytes.

Constructor Specific Documentation:

Create DVB-T energy dispersal.

Parameters: **nsize** – number of blocks.

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

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

declare_sample_delay(dvbt_energy_dispersal_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_energy_dispersal_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_energy_dispersal_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_inner_coder**(*int ninput*, *int noutput*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt_hierarchy_t hierarchy*, *gr::dtv::dvb_code_rate_t coderate*) → dvbt_inner_coder_sptr

Inner coder with Puncturing.

ETSI EN 300 744 Clause 4.3.3Mother convolutional code with rate 1/2. k=1, n=2, K=6. Generator polynomial G1=171(OCT), G2=133(OCT). Punctured to obtain rates of 2/3, 3/4, 5/6, 7/8. Data Input format: Packed bytes (each bit is data). MSB - first, LSB last.Data Output format: 000000X0X1 - QPSK.0000X0X1X2X3 - 16QAM. 00X0X1X2X3X4X5 - 64QAM.

Constructor Specific Documentation:

Create an Inner coder with Puncturing.

Parameters:
- **ninput** – length of input.
- **noutput** – length of output.
- **constellation** – type of constellation.
- **hierarchy** – type of hierarchy used.
- **coderate** – coderate used.

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

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

declare_sample_delay(dvbt_inner_coder_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_inner_coder_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_inner_coder_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_map**(*int nsize*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt_hierarchy_t hierarchy*, *gr::dtv::dvbt_transmission_mode_t transmission*, *float gain*) → dvbt_map_sptr

DVB-T mapper.

ETSI EN 300 744 Clause 4.3.5Data input format: 000000Y0Y1 - QPSK.0000Y0Y1Y2Y3 - 16QAM. 00Y0Y1Y2Y3Y4Y5 - 64QAM. Data output format: complex(real(float), imag(float)).

Constructor Specific Documentation:

Create a DVB-T mapper.

Parameters:
- **nsize** – length of input stream.
- **constellation** – constellation used.
- **hierarchy** – hierarchy used.
- **transmission** – transmission mode used.
- **gain** – gain of complex output stream.

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

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

declare_sample_delay(dvbt_map_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_map_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_map_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_reed_solomon_enc**(*int p*, *int m*, *int gfpoly*, *int n*, *int k*, *int t*, *int s*, *int blocks*) → dvbt_reed_solomon_enc_sptr

Reed Solomon encoder

ETSI EN 300 744 Clause 4.3.2 RS(N=204,K=239,T=8).

Constructor Specific Documentation:

Create a Reed Solomon encoder.

Parameters:
- **p** – characteristic of GF(p^m).
- **m** – we use GF(p^m).
- **gfpoly** – Generator Polynomial.
- **n** – length of codeword of RS coder.
- **k** – length of information sequence of RS encoder.
- **t** – number of corrected errors.
- **s** – shortened length.
- **blocks** – number of blocks to process at once.

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

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

declare_sample_delay(dvbt_reed_solomon_enc_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_reed_solomon_enc_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_reed_solomon_enc_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_reference_signals**(*int itemsize*, *int ninput*, *int noutput*, *gr::dtv::dvb_constellation_t constellation*, *gr::dtv::dvbt_hierarchy_t hierarchy*, *gr::dtv::dvb_code_rate_t code_rate_HP*, *gr::dtv::dvb_code_rate_t code_rate_LP*, *gr::dtv::dvb_guardinterval_t guard_interval*, *gr::dtv::dvbt_transmission_mode_t transmission_mode*, *int include_cell_id*, *int cell_id*) → dvbt_reference_signals_sptr

Reference signals generator.

ETSI EN 300 744 Clause 4.5Data input format: complex(real(float), imag(float)). Data output format: complex(real(float), imag(float)).

Constructor Specific Documentation:

Create Reference signals generator.

Parameters:
- **itemsize** – size of an in/out item.
- **ninput** – input stream length.
- **noutput** – output stream length.
- **constellation** – constellation used.
- **hierarchy** – hierarchy used.
- **code_rate_HP** – high priority stream code rate.
- **code_rate_LP** – low priority stream code rate.
- **guard_interval** – guard interval used.
- **transmission_mode** – transmission mode used.
- **include_cell_id** – include or not Cell ID.
- **cell_id** – value of the Cell ID.

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

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

declare_sample_delay(dvbt_reference_signals_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_reference_signals_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_reference_signals_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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

gnuradio.dtv.**dvbt_symbol_inner_interleaver**(*int ninput*, *gr::dtv::dvbt_transmission_mode_t transmission*, *int direction*) → dvbt_symbol_inner_interleaver_sptr

Symbol interleaver.

ETSI EN 300 744 Clause 4.3.4.2 One block is 12 groups x 126 datawords = 1512 datawords.

Data Input format: 000000I0I1 - QPSK.0000I0I1I2I3 - 16QAM00I0I1I2I3I4I5 - 64QAMData Output format: 000000Y0Y1 - QPSK. 0000Y0Y1Y2Y3 - 16QAM. 00Y0Y1Y2Y3Y4Y5 - 64QAM.

Constructor Specific Documentation:

Create a Symbol interleaver.

Parameters:
- **ninput** – length of input stream.
- **transmission** – transmission mode used
- **direction** – interleave or deinterleave.

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

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

declare_sample_delay(dvbt_symbol_inner_interleaver_sptr self, unsigned int delay)

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

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

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

pc_input_buffers_full_avg(dvbt_symbol_inner_interleaver_sptr self) -> pmt_vector_float

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

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

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

pc_output_buffers_full_avg(dvbt_symbol_inner_interleaver_sptr self) -> pmt_vector_float

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

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

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

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

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

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

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