--- analog_help.txt ---

Help on package gnuradio.analog in gnuradio:

NAME
    gnuradio.analog - Blocks and utilities for analog modulation and demodulation.

PACKAGE CONTENTS
    am_demod
    analog_python
    fm_demod
    fm_emph
    nbfm_rx
    nbfm_tx
    standard_squelch
    wfm_rcv
    wfm_rcv_fmdet
    wfm_rcv_pll
    wfm_tx

SUBMODULES
    analog
    kernel

DATA
    GR_CONST_WAVE = <gr_waveform_t.GR_CONST_WAVE: 100>
    GR_COS_WAVE = <gr_waveform_t.GR_COS_WAVE: 102>
    GR_GAUSSIAN = <noise_type_t.GR_GAUSSIAN: 201>
    GR_IMPULSE = <noise_type_t.GR_IMPULSE: 203>
    GR_LAPLACIAN = <noise_type_t.GR_LAPLACIAN: 202>
    GR_SAW_WAVE = <gr_waveform_t.GR_SAW_WAVE: 105>
    GR_SIN_WAVE = <gr_waveform_t.GR_SIN_WAVE: 101>
    GR_SQR_WAVE = <gr_waveform_t.GR_SQR_WAVE: 103>
    GR_TRI_WAVE = <gr_waveform_t.GR_TRI_WAVE: 104>
    GR_UNIFORM = <noise_type_t.GR_UNIFORM: 200>
    pi = 3.141592653589793

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/analog/__init__.py

--- audio_help.txt ---

Help on package gnuradio.audio in gnuradio:

NAME
    gnuradio.audio

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

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

PACKAGE CONTENTS
    audio_python

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/audio/__init__.py

--- blocks_help.txt ---

Help on package gnuradio.blocks in gnuradio:

NAME
    gnuradio.blocks - Processing blocks common to many flowgraphs.

PACKAGE CONTENTS
    blocks_python
    matrix_interleaver
    msg_meta_to_pair
    msg_pair_to_var
    parse_file_metadata
    pdu_compatibility
    sigmf_sink_minimal
    stream_to_vector_decimator
    var_to_msg

SUBMODULES
    kernel

FUNCTIONS
    count_bits16(...) method of builtins.PyCapsule instance
        count_bits16(x: int) -> int

        return number of set bits in the low 16 bits of x

    count_bits32(...) method of builtins.PyCapsule instance
        count_bits32(x: int) -> int

        return number of set bits in the low 32 bits of x

    count_bits64(...) method of builtins.PyCapsule instance
        count_bits64(x: int) -> int

        return number of set bits in a 64-bit word

    count_bits8(...) method of builtins.PyCapsule instance
        count_bits8(x: int) -> int

        return number of set bits in the low 8 bits of x

    tanhf_lut(...) method of builtins.PyCapsule instance
        tanhf_lut(x: float) -> float

DATA
    FORMAT_DOUBLE = <wavfile_subformat_t.FORMAT_DOUBLE: 7>
    FORMAT_FLAC = <wavfile_format_t.FORMAT_FLAC: 1507328>
    FORMAT_FLOAT = <wavfile_subformat_t.FORMAT_FLOAT: 6>
    FORMAT_OGG = <wavfile_format_t.FORMAT_OGG: 2097152>
    FORMAT_OPUS = <wavfile_subformat_t.FORMAT_OPUS: 100>

FORMAT_PCM_16 = <wavfile_subformat_t.FORMAT_PCM_16: 2>
FORMAT_PCM_24 = <wavfile_subformat_t.FORMAT_PCM_24: 3>
FORMAT_PCM_32 = <wavfile_subformat_t.FORMAT_PCM_32: 4>
FORMAT_PCM_S8 = <wavfile_subformat_t.FORMAT_PCM_S8: 1>
FORMAT_PCM_U8 = <wavfile_subformat_t.FORMAT_PCM_U8: 5>
FORMAT_RF64 = <wavfile_format_t.FORMAT_RF64: 2228224>
FORMAT_VORBIS = <wavfile_subformat_t.FORMAT_VORBIS: 96>
FORMAT_WAV = <wavfile_format_t.FORMAT_WAV: 65536>
GR_FILE_BYTE = <gr_file_types.GR_FILE_BYTE: 0>
GR_FILE_CHAR = <gr_file_types.GR_FILE_BYTE: 0>
GR_FILE_DOUBLE = <gr_file_types.GR_FILE_DOUBLE: 6>
GR_FILE_FLOAT = <gr_file_types.GR_FILE_FLOAT: 5>
GR_FILE_INT = <gr_file_types.GR_FILE_INT: 2>
GR_FILE_LONG = <gr_file_types.GR_FILE_LONG: 3>
GR_FILE_LONG_LONG = <gr_file_types.GR_FILE_LONG_LONG: 4>
GR_FILE_SHORT = <gr_file_types.GR_FILE_SHORT: 1>
METADATA_HEADER_SIZE = 149
METADATA_VERSION = '\x00'
STROBE_EXPONENTIAL = <message_strobe_random_distribution_t.STROBE_EXPO...
STROBE_GAUSSIAN = <message_strobe_random_distribution_t.STROBE_GAUSSIA...
STROBE_POISSON = <message_strobe_random_distribution_t.STROBE_POISSON:...
STROBE_UNIFORM = <message_strobe_random_distribution_t.STROBE_UNIFORM:...

FILE
/usr/local/lib/python3.12/dist-packages/gnuradio/blocks/__init__.py

--- channels_help.txt ---

Help on package gnuradio.channels in gnuradio:

NAME
    gnuradio.channels - Blocks for channel models and related functions.

PACKAGE CONTENTS
    amp_bal
    channels_python
    conj_fs_iqcorr
    distortion_2_gen
    distortion_3_gen
    impairments
    iqbal_gen
    phase_bal
    phase_noise_gen
    quantizer

DATA
    dirname = '/usr/local/lib/python3.12/dist-packages/gnuradio/channels'
    filename = '__init__.py'

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/channels/__init__.py

--- digital_help.txt ---

Help on package gnuradio.digital in gnuradio:

NAME
    gnuradio.digital - Blocks and utilities for digital modulation and demodulation.

PACKAGE CONTENTS
    bpsk
    constellation_map_generator
    cpm
    digital_python
    generic_mod_demod
    gfsk
    gmsk
    modulation_utils
    ofdm_txrx
    packet_utils
    psk
    psk_constellations
    qam
    qam_constellations
    qamlike
    qpsk
    soft_dec_lut_gen
    utils (package)

SUBMODULES
    digital
    mod_codes

CLASSES
        gnuradio.digital.digital_python.cpmmod_bc(gnuradio.gr.gr_python.hier_block2_pb, gnuradio.gr.gr_python.basic_block)
        gmskmod_bc

    class gmskmod_bc(gnuradio.digital.digital_python.cpmmod_bc)
     | gmskmod_bc(samples_per_sym=2, L=4, beta=0.3)
     |
     | Method resolution order:
     | gmskmod_bc
     | gnuradio.digital.digital_python.cpmmod_bc
     | gnuradio.gr.gr_python.hier_block2_pb
     | gnuradio.gr.gr_python.basic_block
     | gnuradio.gr.gr_python.msg_accepter
     | gnuradio.gr.gr_python.messages_msg_accepter
     | pybind11_builtins.pybind11_object
     | builtins.object
     |
     | Methods defined here:
     |

| __init__(self, samples_per_sym=2, L=4, beta=0.3)
| __init__(self: gnuradio.digital.digital_python.cpmmod_bc, type: gnuradio.analog.analog_python.cpm.cpm_type, h: float, samples_per_sym: int, L: int, beta: float = 0.3) -> None
|
| Generic CPM modulator.
|
| Examples:
|
|
|
| The input of this block are symbols from an M-ary alphabet +/-1, +/-3, ..., +/-(M-1). Usually, M = 2 and therefore, the valid inputs are +/-1. The modulator will silently accept any other inputs, though. The output is the phase-modulated signal.
|
| Constructor Specific Documentation:
|
| Make CPM modulator block.
|
| Args:
| type : The modulation type. Can be one of LREC, LRC, LSRC, TFM or GAUSSIAN. See gr_cpm::phase_response() for a detailed description.
| h : The modulation index. is the maximum phase change that can occur between two symbols, i.e., if you only send ones, the phase will increase by every samples. Set this to 0.5 for Minimum Shift Keying variants.
| samples_per_sym : Samples per symbol.
| L : The length of the phase duration in symbols. For L=1, this yields full- response CPM symbols, for L > 1, partial-response.
| beta : For LSRC, this is the rolloff factor. For Gaussian pulses, this is the 3 dB time-bandwidth product.
|
| --
| Data descriptors defined here:
|
| __dict__
| dictionary for instance variables
|
| --
| Methods inherited from gnuradio.digital.digital_python.cpmmod_bc:
|
| beta(...)
| beta(self: gnuradio.digital.digital_python.cpmmod_bc) -> float
|
| Return the value of beta for the modulator.
|
| index(...)
| index(self: gnuradio.digital.digital_python.cpmmod_bc) -> float
|
| Return the modulation index of the modulator.
|
| samples_per_sym(...)
| samples_per_sym(self: gnuradio.digital.digital_python.cpmmod_bc) -> int
|

```
 |  Return the number of samples per symbol.
 |
 | taps(...)
 |  taps(self: gnuradio.digital.digital_python.cpmmod_bc) -> List[float]
 |
 |  Return the phase response FIR taps.
 |
 | type(...)
 |  type(self: gnuradio.digital.digital_python.cpmmod_bc) -> int
 |
 |  Return the type of CPM modulator.
 |
 | ----------------------------------------------------------------------
 | Static methods inherited from gnuradio.digital.digital_python.cpmmod_bc:
 |
 | make_gmskmod_bc(...) method of builtins.PyCapsule instance
 |  make_gmskmod_bc(samples_per_sym: int = 2, L: int = 4, beta: float = 0.3) -> gnuradio.digital.digital_python.cpmmod_bc
 |
 |  Make GMSK modulator block.
 |
 |
 |  The type is GAUSSIAN and the modulation index for GMSK is 0.5. This are populated automatically by this factory function.
 |
 | ----------------------------------------------------------------------
 | Methods inherited from gnuradio.gr.gr_python.hier_block2_pb:
 |
 | all_max_output_buffer_p(...)
 |  all_max_output_buffer_p(self: gnuradio.gr.gr_python.hier_block2_pb) -> bool
 |
 | all_min_output_buffer_p(...)
 |  all_min_output_buffer_p(self: gnuradio.gr.gr_python.hier_block2_pb) -> bool
 |
 | disconnect_all(...)
 |  disconnect_all(self: gnuradio.gr.gr_python.hier_block2_pb) -> None
 |
 | has_msg_port(...)
 |  has_msg_port(self: gnuradio.gr.gr_python.hier_block2_pb, which_port: pmt.pmt_python.pmt_base) -> bool
 |
 | lock(...)
 |  lock(self: gnuradio.gr.gr_python.hier_block2_pb) -> None
 |
 | log_level(...)
 |  log_level(self: gnuradio.gr.gr_python.hier_block2_pb) -> str
 |
 | max_output_buffer(...)
 |  max_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb, port: int = 0) -> int
 |
 | message_port_is_hier(...)
 |  message_port_is_hier(self: gnuradio.gr.gr_python.hier_block2_pb, port_id: pmt.pmt_python.pmt_base) -> bool
```

```
    |
    |  message_port_is_hier_in(...)
    |      message_port_is_hier_in(self: gnuradio.gr.gr_python.hier_block2_pb, port_id:
pmt.pmt_python.pmt_base) -> bool
    |
    |  message_port_is_hier_out(...)
    |      message_port_is_hier_out(self: gnuradio.gr.gr_python.hier_block2_pb, port_id:
pmt.pmt_python.pmt_base) -> bool
    |
    |  min_output_buffer(...)
    |      min_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb, port: int = 0) -> int
    |
    |  primitive_connect(...)
    |      primitive_connect(*args, **kwargs)
    |      Overloaded function.
    |
    |          1. primitive_connect(self: gnuradio.gr.gr_python.hier_block2_pb, block:
gnuradio.gr.gr_python.basic_block) -> None
    |
    |          2. primitive_connect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, src_port: int, dst: gnuradio.gr.gr_python.basic_block,
dst_port: int) -> None
    |
    |  primitive_disconnect(...)
    |      primitive_disconnect(*args, **kwargs)
    |      Overloaded function.
    |
    |          1. primitive_disconnect(self: gnuradio.gr.gr_python.hier_block2_pb, block:
gnuradio.gr.gr_python.basic_block) -> None
    |
    |          2. primitive_disconnect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, src_port: int, dst: gnuradio.gr.gr_python.basic_block,
dst_port: int) -> None
    |
    |  primitive_message_port_register_hier_in(...)
    |      primitive_message_port_register_hier_in(self: gnuradio.gr.gr_python.hier_block2_pb,
port_id: pmt.pmt_python.pmt_base) -> None
    |
    |  primitive_message_port_register_hier_out(...)
    |      primitive_message_port_register_hier_out(self: gnuradio.gr.gr_python.hier_block2_pb,
port_id: pmt.pmt_python.pmt_base) -> None
    |
    |  primitive_msg_connect(...)
    |      primitive_msg_connect(*args, **kwargs)
    |      Overloaded function.
    |
    |          1. primitive_msg_connect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, srcport: pmt.pmt_python.pmt_base, dst:
gnuradio.gr.gr_python.basic_block, dstport: pmt.pmt_python.pmt_base) -> None
    |
    |          2. primitive_msg_connect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, srcport: str, dst: gnuradio.gr.gr_python.basic_block,
dstport: str) -> None
```

```
    |
    | primitive_msg_disconnect(...)
    |     primitive_msg_disconnect(*args, **kwargs)
    |     Overloaded function.
    |
        |           1. primitive_msg_disconnect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, srcport: pmt.pmt_python.pmt_base, dst:
gnuradio.gr.gr_python.basic_block, dstport: pmt.pmt_python.pmt_base) -> None
    |
        |           2. primitive_msg_disconnect(self: gnuradio.gr.gr_python.hier_block2_pb, src:
gnuradio.gr.gr_python.basic_block, srcport: str, dst: gnuradio.gr.gr_python.basic_block,
dstport: str) -> None
    |
    | processor_affinity(...)
    |     processor_affinity(self: gnuradio.gr.gr_python.hier_block2_pb) -> List[int]
    |
    | self(...)
    |     self(self: gnuradio.gr.gr_python.hier_block2_pb) -> gnuradio.gr.gr_python.basic_block
    |
    | set_log_level(...)
    |     set_log_level(self: gnuradio.gr.gr_python.hier_block2_pb, level: str) -> None
    |
    | set_max_output_buffer(...)
    |     set_max_output_buffer(*args, **kwargs)
    |     Overloaded function.
    |
          |             1. set_max_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb,
max_output_buffer: int) -> None
    |
        |           2. set_max_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb, port: int,
max_output_buffer: int) -> None
    |
    | set_min_output_buffer(...)
    |     set_min_output_buffer(*args, **kwargs)
    |     Overloaded function.
    |
          |             1. set_min_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb,
min_output_buffer: int) -> None
    |
        |           2. set_min_output_buffer(self: gnuradio.gr.gr_python.hier_block2_pb, port: int,
min_output_buffer: int) -> None
    |
    | set_processor_affinity(...)
     |       set_processor_affinity(self: gnuradio.gr.gr_python.hier_block2_pb, mask: List[int]) ->
None
    |
    | to_hier_block2(...)
              |                 to_hier_block2(self: gnuradio.gr.gr_python.hier_block2_pb) ->
gnuradio.gr.gr_python.hier_block2_pb
    |
    | unlock(...)
    |     unlock(self: gnuradio.gr.gr_python.hier_block2_pb) -> None
    |
```

```
 |  unset_processor_affinity(...)
 |      unset_processor_affinity(self: gnuradio.gr.gr_python.hier_block2_pb) -> None
 |
 |  ----------------------------------------------------------------------
 |  Methods inherited from gnuradio.gr.gr_python.basic_block:
 |
 |  alias(...)
 |      alias(self: gnuradio.gr.gr_python.basic_block) -> str
 |
 |  alias_pmt(...)
 |      alias_pmt(self: gnuradio.gr.gr_python.basic_block) -> pmt.pmt_python.pmt_base
 |
 |  alias_set(...)
 |      alias_set(self: gnuradio.gr.gr_python.basic_block) -> bool
 |
 |  check_topology(...)
 |      check_topology(self: gnuradio.gr.gr_python.basic_block, ninputs: int, noutputs: int) ->
bool
 |
 |  delete_head_nowait(...)
 |          delete_head_nowait(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base) -> pmt.pmt_python.pmt_base
 |
 |  empty_handled_p(...)
 |      empty_handled_p(*args, **kwargs)
 |      Overloaded function.
 |
 |          1. empty_handled_p(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base) -> bool
 |
 |      2. empty_handled_p(self: gnuradio.gr.gr_python.basic_block) -> bool
 |
 |  empty_p(...)
 |      empty_p(*args, **kwargs)
 |      Overloaded function.
 |
 |          1. empty_p(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base) -> bool
 |
 |      2. empty_p(self: gnuradio.gr.gr_python.basic_block) -> bool
 |
 |  erase_msg(...)
 |          erase_msg(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base, it: std::_Deque_iterator<std::shared_ptr<pmt::pmt_base>,
std::shared_ptr<pmt::pmt_base>&, std::shared_ptr<pmt::pmt_base>*>) -> None
 |
 |  get_iterator(...)
 |          get_iterator(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base) -> std::_Deque_iterator<std::shared_ptr<pmt::pmt_base>,
std::shared_ptr<pmt::pmt_base>&, std::shared_ptr<pmt::pmt_base>*>
 |
 |  get_msg_map(...)
 |          get_msg_map(self: gnuradio.gr.gr_python.basic_block) ->
```

```
Dict[pmt.pmt_python.pmt_base, List[pmt.pmt_python.pmt_base]]
    |
    |  identifier(...)
    |      identifier(self: gnuradio.gr.gr_python.basic_block) -> str
    |
    |  input_signature(...)
    |                  input_signature(self: gnuradio.gr.gr_python.basic_block) ->
gnuradio.gr.gr_python.io_signature
    |
    |  insert_tail(...)
    |                  insert_tail(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base, msg: pmt.pmt_python.pmt_base) -> None
    |
    |  message_port_pub(...)
    |                  message_port_pub(self: gnuradio.gr.gr_python.basic_block, port_id:
pmt.pmt_python.pmt_base, msg: pmt.pmt_python.pmt_base) -> None
    |
    |  message_port_register_in(...)
    |                  message_port_register_in(self: gnuradio.gr.gr_python.basic_block, port_id:
pmt.pmt_python.pmt_base) -> None
    |
    |  message_port_register_out(...)
    |                  message_port_register_out(self: gnuradio.gr.gr_python.basic_block, port_id:
pmt.pmt_python.pmt_base) -> None
    |
    |  message_port_sub(...)
    |                  message_port_sub(self: gnuradio.gr.gr_python.basic_block, port_id:
pmt.pmt_python.pmt_base, target: pmt.pmt_python.pmt_base) -> None
    |
    |  message_port_unsub(...)
    |                  message_port_unsub(self: gnuradio.gr.gr_python.basic_block, port_id:
pmt.pmt_python.pmt_base, target: pmt.pmt_python.pmt_base) -> None
    |
    |  message_ports_in(...)
    |                  message_ports_in(self: gnuradio.gr.gr_python.basic_block) ->
pmt.pmt_python.pmt_base
    |
    |  message_ports_out(...)
    |                  message_ports_out(self: gnuradio.gr.gr_python.basic_block) ->
pmt.pmt_python.pmt_base
    |
    |  message_subscribers(...)
    |                  message_subscribers(self: gnuradio.gr.gr_python.basic_block, port:
pmt.pmt_python.pmt_base) -> pmt.pmt_python.pmt_base
    |
    |  name(...)
    |      name(self: gnuradio.gr.gr_python.basic_block) -> str
    |
    |  nmsgs(...)
    |                  nmsgs(self: gnuradio.gr.gr_python.basic_block, which_port:
pmt.pmt_python.pmt_base) -> int
    |
    |  output_signature(...)
```

```
 |  output_signature(self: gnuradio.gr.gr_python.basic_block) -> gnuradio.gr.gr_python.io_signature
 |
 |  set_block_alias(...)
 |      set_block_alias(self: gnuradio.gr.gr_python.basic_block, name: str) -> None
 |
 |  symbol_name(...)
 |      symbol_name(self: gnuradio.gr.gr_python.basic_block) -> str
 |
 |  symbolic_id(...)
 |      symbolic_id(self: gnuradio.gr.gr_python.basic_block) -> int
 |
 |  to_basic_block(...)
 |      to_basic_block(self: gnuradio.gr.gr_python.basic_block) -> gnuradio.gr.gr_python.basic_block
 |
 |  unique_id(...)
 |      unique_id(self: gnuradio.gr.gr_python.basic_block) -> int
 |
 |  ----------------------------------------------------------------------
 |  Methods inherited from gnuradio.gr.gr_python.msg_accepter:
 |
 |  post(...)
 |      post(self: gnuradio.gr.gr_python.msg_accepter, which_port: pmt.pmt_python.pmt_base, msg: pmt.pmt_python.pmt_base) -> None
 |
 |  ----------------------------------------------------------------------
 |  Static methods inherited from pybind11_builtins.pybind11_object:
 |
 |  __new__(*args, **kwargs) class method of pybind11_builtins.pybind11_object
 |      Create and return a new object.  See help(type) for accurate signature.

FUNCTIONS
    exp(z, /)
        Return the exponential value e**z.

    ln = log(...)
        log(x, [base=math.e])
        Return the logarithm of x to the given base.

        If the base is not specified, returns the natural logarithm (base e) of x.

    log(...)
        log(x, [base=math.e])
        Return the logarithm of x to the given base.

        If the base is not specified, returns the natural logarithm (base e) of x.

    modulate_vector_bc(...) method of builtins.PyCapsule instance
        modulate_vector_bc(modulator: gnuradio.gr.gr_python.basic_block, data: List[int], taps: List[float]) -> List[complex]

    sqrt(x, /)
```

Return the square root of x.

DATA
CMA = <adaptive_algorithm_t.CMA: 2>
DIFF_DIFFERENTIAL = <diff_coding_type.DIFF_DIFFERENTIAL: 0>
DIFF_NRZI = <diff_coding_type.DIFF_NRZI: 1>
EVM_DB = <evm_measurement_t.EVM_DB: 1>
EVM_PERCENT = <evm_measurement_t.EVM_PERCENT: 0>
IR_MMSE_8TAP = <ir_type.IR_MMSE_8TAP: 0>
IR_NONE = <ir_type.IR_NONE: -1>
IR_PFB_MF = <ir_type.IR_PFB_MF: 2>
IR_PFB_NO_MF = <ir_type.IR_PFB_NO_MF: 1>
LMS = <adaptive_algorithm_t.LMS: 0>
NLMS = <adaptive_algorithm_t.NLMS: 1>
SNR_EST_M2M4 = <snr_est_type_t.SNR_EST_M2M4: 2>
SNR_EST_SIMPLE = <snr_est_type_t.SNR_EST_SIMPLE: 0>
SNR_EST_SKEW = <snr_est_type_t.SNR_EST_SKEW: 1>
SNR_EST_SVR = <snr_est_type_t.SNR_EST_SVR: 3>
TED_DANDREA_AND_MENGALI_GEN_MSK = <ted_type.TED_DANDREA_AND_MENGALI_GE...
TED_EARLY_LATE = <ted_type.TED_EARLY_LATE: 5>
TED_GARDNER = <ted_type.TED_GARDNER: 4>
TED_MENGALI_AND_DANDREA_GMSK = <ted_type.TED_MENGALI_AND_DANDREA_GMSK:...
TED_MOD_MUELLER_AND_MULLER = <ted_type.TED_MOD_MUELLER_AND_MULLER: 1>
TED_MUELLER_AND_MULLER = <ted_type.TED_MUELLER_AND_MULLER: 0>
TED_NONE = <ted_type.TED_NONE: -1>
TED_SIGNAL_TIMES_SLOPE_ML = <ted_type.TED_SIGNAL_TIMES_SLOPE_ML: 7>
TED_SIGNUM_TIMES_SLOPE_ML = <ted_type.TED_SIGNUM_TIMES_SLOPE_ML: 8>
TED_ZERO_CROSSING = <ted_type.TED_ZERO_CROSSING: 2>
THRESHOLD_ABSOLUTE = <tm_type.THRESHOLD_ABSOLUTE: 1>
THRESHOLD_DYNAMIC = <tm_type.THRESHOLD_DYNAMIC: 0>
TRELLIS_EUCLIDEAN = <trellis_metric_type_t.TRELLIS_EUCLIDEAN: 200>
TRELLIS_HARD_BIT = <trellis_metric_type_t.TRELLIS_HARD_BIT: 202>
TRELLIS_HARD_SYMBOL = <trellis_metric_type_t.TRELLIS_HARD_SYMBOL: 201>
pi = 3.141592653589793
shared_demod_args = '   samples_per_symbol: samples per baud >= 2 (f....
shared_mod_args = '   samples_per_symbol: samples per baud >= 2 (f.....

FILE
/usr/local/lib/python3.12/dist-packages/gnuradio/digital/__init__.py

--- dtv_help.txt ---

Help on package gnuradio.dtv in gnuradio:

NAME
    gnuradio.dtv - Blocks and utilities for digital TV module.

PACKAGE CONTENTS
    atsc_rx
    atsc_rx_filter
    dtv_python

SUBMODULES
    dtv

DATA
    ALPHA1 = <dvbt_hierarchy_t.ALPHA1: 1>
    ALPHA2 = <dvbt_hierarchy_t.ALPHA2: 2>
    ALPHA4 = <dvbt_hierarchy_t.ALPHA4: 3>
    ATSC_CHANNEL_BW = 6000000.0
    ATSC_RRC_SYMS = 8
    ATSC_SYMBOL_RATE = 10762237.762237763
    BANDWIDTH_10_0_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_10_0_MHZ: 5>
    BANDWIDTH_1_7_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_1_7_MHZ: 0>
    BANDWIDTH_5_0_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_5_0_MHZ: 1>
    BANDWIDTH_6_0_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_6_0_MHZ: 2>
    BANDWIDTH_7_0_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_7_0_MHZ: 3>
    BANDWIDTH_8_0_MHZ = <dvbt2_bandwidth_t.BANDWIDTH_8_0_MHZ: 4>
    C100_180 = <dvb_code_rate_t.C100_180: 17>
    C104_180 = <dvb_code_rate_t.C104_180: 18>
    C116_180 = <dvb_code_rate_t.C116_180: 23>
    C11_20 = <dvb_code_rate_t.C11_20: 16>
    C11_45 = <dvb_code_rate_t.C11_45: 35>
    C11_45_MEDIUM = <dvb_code_rate_t.C11_45_MEDIUM: 43>
    C11_45_VLSNR_SF2 = <dvb_code_rate_t.C11_45_VLSNR_SF2: 46>
    C124_180 = <dvb_code_rate_t.C124_180: 25>
    C128_180 = <dvb_code_rate_t.C128_180: 27>
    C132_180 = <dvb_code_rate_t.C132_180: 29>
    C135_180 = <dvb_code_rate_t.C135_180: 31>
    C13_18 = <dvb_code_rate_t.C13_18: 28>
    C13_45 = <dvb_code_rate_t.C13_45: 12>
    C140_180 = <dvb_code_rate_t.C140_180: 32>
    C14_45 = <dvb_code_rate_t.C14_45: 37>
    C154_180 = <dvb_code_rate_t.C154_180: 34>
    C18_30 = <dvb_code_rate_t.C18_30: 20>
    C1_2 = <dvb_code_rate_t.C1_2: 3>
    C1_3 = <dvb_code_rate_t.C1_3: 1>
    C1_3_MEDIUM = <dvb_code_rate_t.C1_3_MEDIUM: 44>
    C1_3_VLSNR = <dvb_code_rate_t.C1_3_VLSNR: 49>
    C1_4 = <dvb_code_rate_t.C1_4: 0>
    C1_5_MEDIUM = <dvb_code_rate_t.C1_5_MEDIUM: 42>

C1_5_VLSNR = <dvb_code_rate_t.C1_5_VLSNR: 47>
C1_5_VLSNR_SF2 = <dvb_code_rate_t.C1_5_VLSNR_SF2: 45>
C20_30 = <dvb_code_rate_t.C20_30: 24>
C22_30 = <dvb_code_rate_t.C22_30: 30>
C23_36 = <dvb_code_rate_t.C23_36: 22>
C25_36 = <dvb_code_rate_t.C25_36: 26>
C26_45 = <dvb_code_rate_t.C26_45: 19>
C28_45 = <dvb_code_rate_t.C28_45: 21>
C2_3 = <dvb_code_rate_t.C2_3: 5>
C2_5 = <dvb_code_rate_t.C2_5: 2>
C2_9_VLSNR = <dvb_code_rate_t.C2_9_VLSNR: 41>
C32_45 = <dvb_code_rate_t.C32_45: 40>
C3_4 = <dvb_code_rate_t.C3_4: 6>
C3_5 = <dvb_code_rate_t.C3_5: 4>
C4_15 = <dvb_code_rate_t.C4_15: 36>
C4_15_VLSNR = <dvb_code_rate_t.C4_15_VLSNR: 48>
C4_5 = <dvb_code_rate_t.C4_5: 7>
C5_6 = <dvb_code_rate_t.C5_6: 8>
C7_15 = <dvb_code_rate_t.C7_15: 38>
C7_8 = <dvb_code_rate_t.C7_8: 9>
C7_9 = <dvb_code_rate_t.C7_9: 33>
C8_15 = <dvb_code_rate_t.C8_15: 39>
C8_9 = <dvb_code_rate_t.C8_9: 10>
C90_180 = <dvb_code_rate_t.C90_180: 14>
C96_180 = <dvb_code_rate_t.C96_180: 15>
C9_10 = <dvb_code_rate_t.C9_10: 11>
C9_20 = <dvb_code_rate_t.C9_20: 13>
CARRIERS_EXTENDED = <dvbt2_extended_carrier_t.CARRIERS_EXTENDED: 1>
CARRIERS_NORMAL = <dvbt2_extended_carrier_t.CARRIERS_NORMAL: 0>
CATV_MOD_256QAM = <catv_constellation_t.CATV_MOD_256QAM: 1>
CATV_MOD_64QAM = <catv_constellation_t.CATV_MOD_64QAM: 0>
C_OTHER = <dvb_code_rate_t.C_OTHER: 50>
EQUALIZATION_OFF = <dvbt2_equalization_t.EQUALIZATION_OFF: 0>
EQUALIZATION_ON = <dvbt2_equalization_t.EQUALIZATION_ON: 1>
FECFRAME_MEDIUM = <dvb_framesize_t.FECFRAME_MEDIUM: 2>
FECFRAME_NORMAL = <dvb_framesize_t.FECFRAME_NORMAL: 1>
FECFRAME_SHORT = <dvb_framesize_t.FECFRAME_SHORT: 0>
FFTSIZE_16K = <dvbt2_fftsize_t.FFTSIZE_16K: 4>
FFTSIZE_16K_T2GI = <dvbt2_fftsize_t.FFTSIZE_16K_T2GI: 11>
FFTSIZE_1K = <dvbt2_fftsize_t.FFTSIZE_1K: 3>
FFTSIZE_2K = <dvbt2_fftsize_t.FFTSIZE_2K: 0>
FFTSIZE_32K = <dvbt2_fftsize_t.FFTSIZE_32K: 5>
FFTSIZE_32K_T2GI = <dvbt2_fftsize_t.FFTSIZE_32K_T2GI: 7>
FFTSIZE_4K = <dvbt2_fftsize_t.FFTSIZE_4K: 2>
FFTSIZE_8K = <dvbt2_fftsize_t.FFTSIZE_8K: 1>
FFTSIZE_8K_T2GI = <dvbt2_fftsize_t.FFTSIZE_8K_T2GI: 6>
GI_19_128 = <dvb_guardinterval_t.GI_19_128: 5>
GI_19_256 = <dvb_guardinterval_t.GI_19_256: 6>
GI_1_128 = <dvb_guardinterval_t.GI_1_128: 4>
GI_1_16 = <dvb_guardinterval_t.GI_1_16: 1>
GI_1_32 = <dvb_guardinterval_t.GI_1_32: 0>
GI_1_4 = <dvb_guardinterval_t.GI_1_4: 3>
GI_1_8 = <dvb_guardinterval_t.GI_1_8: 2>

INBAND_OFF = <dvbt2_inband_t.INBAND_OFF: 0>
INBAND_ON = <dvbt2_inband_t.INBAND_ON: 1>
INPUTMODE_HIEFF = <dvbt2_inputmode_t.INPUTMODE_HIEFF: 1>
INPUTMODE_NORMAL = <dvbt2_inputmode_t.INPUTMODE_NORMAL: 0>
INTERPOLATION_OFF = <dvbs2_interpolation_t.INTERPOLATION_OFF: 0>
INTERPOLATION_ON = <dvbs2_interpolation_t.INTERPOLATION_ON: 1>
L1_MOD_16QAM = <dvbt2_l1constellation_t.L1_MOD_16QAM: 2>
L1_MOD_64QAM = <dvbt2_l1constellation_t.L1_MOD_64QAM: 3>
L1_MOD_BPSK = <dvbt2_l1constellation_t.L1_MOD_BPSK: 0>
L1_MOD_QPSK = <dvbt2_l1constellation_t.L1_MOD_QPSK: 1>
L1_SCRAMBLED_OFF = <dvbt2_l1scrambled_t.L1_SCRAMBLED_OFF: 0>
L1_SCRAMBLED_ON = <dvbt2_l1scrambled_t.L1_SCRAMBLED_ON: 1>
MISO_TX1 = <dvbt2_misogroup_t.MISO_TX1: 0>
MISO_TX2 = <dvbt2_misogroup_t.MISO_TX2: 1>
MOD_128APSK = <dvb_constellation_t.MOD_128APSK: 14>
MOD_16APSK = <dvb_constellation_t.MOD_16APSK: 6>
MOD_16QAM = <dvb_constellation_t.MOD_16QAM: 1>
MOD_256APSK = <dvb_constellation_t.MOD_256APSK: 15>
MOD_256QAM = <dvb_constellation_t.MOD_256QAM: 3>
MOD_32APSK = <dvb_constellation_t.MOD_32APSK: 8>
MOD_4_12_16APSK = <dvb_constellation_t.MOD_4_12_16APSK: 9>
MOD_4_12_20_28APSK = <dvb_constellation_t.MOD_4_12_20_28APSK: 13>
MOD_4_8_4_16APSK = <dvb_constellation_t.MOD_4_8_4_16APSK: 10>
MOD_64APSK = <dvb_constellation_t.MOD_64APSK: 11>
MOD_64QAM = <dvb_constellation_t.MOD_64QAM: 2>
MOD_8APSK = <dvb_constellation_t.MOD_8APSK: 5>
MOD_8PSK = <dvb_constellation_t.MOD_8PSK: 4>
MOD_8VSB = <dvb_constellation_t.MOD_8VSB: 18>
MOD_8_16_20_20APSK = <dvb_constellation_t.MOD_8_16_20_20APSK: 12>
MOD_8_8APSK = <dvb_constellation_t.MOD_8_8APSK: 7>
MOD_BPSK = <dvb_constellation_t.MOD_BPSK: 16>
MOD_BPSK_SF2 = <dvb_constellation_t.MOD_BPSK_SF2: 17>
MOD_OTHER = <dvb_constellation_t.MOD_OTHER: 19>
MOD_QPSK = <dvb_constellation_t.MOD_QPSK: 0>
NH = <dvbt_hierarchy_t.NH: 0>
PAPR_ACE = <dvbt2_papr_t.PAPR_ACE: 1>
PAPR_BOTH = <dvbt2_papr_t.PAPR_BOTH: 3>
PAPR_OFF = <dvbt2_papr_t.PAPR_OFF: 0>
PAPR_TR = <dvbt2_papr_t.PAPR_TR: 2>
PILOTS_OFF = <dvbs2_pilots_t.PILOTS_OFF: 0>
PILOTS_ON = <dvbs2_pilots_t.PILOTS_ON: 1>
PILOT_PP1 = <dvbt2_pilotpattern_t.PILOT_PP1: 0>
PILOT_PP2 = <dvbt2_pilotpattern_t.PILOT_PP2: 1>
PILOT_PP3 = <dvbt2_pilotpattern_t.PILOT_PP3: 2>
PILOT_PP4 = <dvbt2_pilotpattern_t.PILOT_PP4: 3>
PILOT_PP5 = <dvbt2_pilotpattern_t.PILOT_PP5: 4>
PILOT_PP6 = <dvbt2_pilotpattern_t.PILOT_PP6: 5>
PILOT_PP7 = <dvbt2_pilotpattern_t.PILOT_PP7: 6>
PILOT_PP8 = <dvbt2_pilotpattern_t.PILOT_PP8: 7>
PREAMBLE_NON_T2 = <dvbt2_preamble_t.PREAMBLE_NON_T2: 2>
PREAMBLE_T2_LITE_MISO = <dvbt2_preamble_t.PREAMBLE_T2_LITE_MISO: 4>
PREAMBLE_T2_LITE_SISO = <dvbt2_preamble_t.PREAMBLE_T2_LITE_SISO: 3>
PREAMBLE_T2_MISO = <dvbt2_preamble_t.PREAMBLE_T2_MISO: 1>

PREAMBLE_T2_SISO = <dvbt2_preamble_t.PREAMBLE_T2_SISO: 0>
RESERVED_OFF = <dvbt2_reservedbiasbits_t.RESERVED_OFF: 0>
RESERVED_ON = <dvbt2_reservedbiasbits_t.RESERVED_ON: 1>
ROTATION_OFF = <dvbt2_rotation_t.ROTATION_OFF: 0>
ROTATION_ON = <dvbt2_rotation_t.ROTATION_ON: 1>
RO_0_05 = <dvbs2_rolloff_factor_t.RO_0_05: 6>
RO_0_10 = <dvbs2_rolloff_factor_t.RO_0_10: 5>
RO_0_15 = <dvbs2_rolloff_factor_t.RO_0_15: 4>
RO_0_20 = <dvbs2_rolloff_factor_t.RO_0_20: 2>
RO_0_25 = <dvbs2_rolloff_factor_t.RO_0_25: 1>
RO_0_35 = <dvbs2_rolloff_factor_t.RO_0_35: 0>
RO_RESERVED = <dvbs2_rolloff_factor_t.RO_RESERVED: 3>
SHOWLEVELS_OFF = <dvbt2_showlevels_t.SHOWLEVELS_OFF: 0>
SHOWLEVELS_ON = <dvbt2_showlevels_t.SHOWLEVELS_ON: 1>
STANDARD_DVBS2 = <dvb_standard_t.STANDARD_DVBS2: 0>
STANDARD_DVBT2 = <dvb_standard_t.STANDARD_DVBT2: 1>
STREAMTYPE_BOTH = <dvbt2_streamtype_t.STREAMTYPE_BOTH: 2>
STREAMTYPE_GS = <dvbt2_streamtype_t.STREAMTYPE_GS: 1>
STREAMTYPE_TS = <dvbt2_streamtype_t.STREAMTYPE_TS: 0>
T2k = <dvbt_transmission_mode_t.T2k: 0>
T8k = <dvbt_transmission_mode_t.T8k: 1>
VERSION_111 = <dvbt2_version_t.VERSION_111: 0>
VERSION_121 = <dvbt2_version_t.VERSION_121: 1>
VERSION_131 = <dvbt2_version_t.VERSION_131: 2>

FILE
/usr/local/lib/python3.12/dist-packages/gnuradio/dtv/__init__.py

--- fec_help.txt ---

Help on package gnuradio.fec in gnuradio:

NAME
    gnuradio.fec - Blocks for forward error correction.

PACKAGE CONTENTS
    LDPC (package)
    bercurve_generator
    bitflip
    capillary_threaded_decoder
    capillary_threaded_encoder
    extended_async_encoder
    extended_decoder
    extended_encoder
    extended_tagged_decoder
    extended_tagged_encoder
    fec_python
    fec_test
    polar (package)
    threaded_decoder
    threaded_encoder

SUBMODULES
    code

FUNCTIONS
    cc_decoder_make = make(...) method of builtins.PyCapsule instance
        make(frame_size: int, k: int, rate: int, polys: List[int], start_state: int = 0, end_state: int = -1, mode: gnuradio.fec.fec_python._cc_mode_t = <_cc_mode_t.CC_STREAMING: 0>, padded: bool = False) -> gnuradio.fec.fec_python.generic_decoder

    cc_encoder_make = make(...) method of builtins.PyCapsule instance
            make(frame_size: int, k: int, rate: int, polys: List[int], start_state: int = 0, mode: gnuradio.fec.fec_python._cc_mode_t = <_cc_mode_t.CC_STREAMING: 0>, padded: bool = False) -> gnuradio.fec.fec_python.generic_encoder

    ccsds_encoder_make = make(...) method of builtins.PyCapsule instance
        make(frame_size: int, start_state: int = 0, mode: gnuradio.fec.fec_python._cc_mode_t = <_cc_mode_t.CC_STREAMING: 0>) -> gnuradio.fec.fec_python.generic_encoder

    dummy_decoder_make = make(...) method of builtins.PyCapsule instance
       make(frame_size: int) -> gnuradio.fec.fec_python.generic_decoder

    dummy_encoder_make = make(...) method of builtins.PyCapsule instance
                make(frame_size: int, pack: bool = False, packed_bits: bool = False) -> gnuradio.fec.fec_python.generic_encoder

    get_decoder_input_conversion(...) method of builtins.PyCapsule instance
        get_decoder_input_conversion(my_decoder: gnuradio.fec.fec_python.generic_decoder)

-> str

    get_decoder_input_item_size(...) method of builtins.PyCapsule instance
        get_decoder_input_item_size(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> int

    get_decoder_input_size(...) method of builtins.PyCapsule instance
        get_decoder_input_size(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> int

    get_decoder_output_conversion(...) method of builtins.PyCapsule instance
                                                    get_decoder_output_conversion(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> str

    get_decoder_output_item_size(...) method of builtins.PyCapsule instance
        get_decoder_output_item_size(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> int

    get_decoder_output_size(...) method of builtins.PyCapsule instance
        get_decoder_output_size(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> int

    get_encoder_input_conversion(...) method of builtins.PyCapsule instance
        get_encoder_input_conversion(my_encoder: gnuradio.fec.fec_python.generic_encoder) -> str

    get_encoder_input_size(...) method of builtins.PyCapsule instance
        get_encoder_input_size(my_encoder: gnuradio.fec.fec_python.generic_encoder) -> int

    get_encoder_output_conversion(...) method of builtins.PyCapsule instance
        get_encoder_output_conversion(my_encoder: gnuradio.fec.fec_python.generic_encoder) -> str

    get_encoder_output_size(...) method of builtins.PyCapsule instance
        get_encoder_output_size(my_encoder: gnuradio.fec.fec_python.generic_encoder) -> int

    get_history(...) method of builtins.PyCapsule instance
        get_history(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> int

    get_shift(...) method of builtins.PyCapsule instance
        get_shift(my_decoder: gnuradio.fec.fec_python.generic_decoder) -> float

    ldpc_decoder_make = make(...) method of builtins.PyCapsule instance
                                    make(alist_file: str, max_iterations: int = 50) -> gnuradio.fec.fec_python.generic_decoder

    ldpc_encoder_make = make(...) method of builtins.PyCapsule instance
        make(alist_file: str) -> gnuradio.fec.fec_python.generic_encoder

    ldpc_gen_mtrx_encoder_make = make(...) method of builtins.PyCapsule instance
                                    make(G_obj: gnuradio.fec.fec_python.code.ldpc_G_matrix) -> gnuradio.fec.fec_python.generic_encoder

    ldpc_par_mtrx_encoder_make = make(...) method of builtins.PyCapsule instance
        make(alist_file: str, gap: int = 0) -> gnuradio.fec.fec_python.generic_encoder

ldpc_par_mtrx_encoder_make_H = make_H(...) method of builtins.PyCapsule instance
make_H(H_obj: gnuradio.fec.fec_python.code.ldpc_H_matrix) -> gnuradio.fec.fec_python.generic_encoder

polar_encoder_make = make(...) method of builtins.PyCapsule instance
make(block_size: int, num_info_bits: int, frozen_bit_positions: List[int], frozen_bit_values: List[int], is_packed: bool = False) -> gnuradio.fec.fec_python.generic_encoder

polar_encoder_systematic_make = make(...) method of builtins.PyCapsule instance
make(block_size: int, num_info_bits: int, frozen_bit_positions: List[int]) -> gnuradio.fec.fec_python.generic_encoder

repetition_decoder_make = make(...) method of builtins.PyCapsule instance
make(frame_size: int, rep: int, ap_prob: float = 0.5) -> gnuradio.fec.fec_python.generic_decoder

repetition_encoder_make = make(...) method of builtins.PyCapsule instance
make(frame_size: int, rep: int) -> gnuradio.fec.fec_python.generic_encoder

tpc_decoder_make = make(...) method of builtins.PyCapsule instance
make(row_poly: List[int], col_poly: List[int], krow: int, kcol: int, bval: int, qval: int, max_iter: int, decoder_type: int) -> gnuradio.fec.fec_python.generic_decoder

tpc_encoder_make = make(...) method of builtins.PyCapsule instance
make(row_poly: List[int], col_poly: List[int], krow: int, kcol: int, bval: int, qval: int) -> gnuradio.fec.fec_python.generic_encoder

DATA
CC_STREAMING = <_cc_mode_t.CC_STREAMING: 0>
CC_TAILBITING = <_cc_mode_t.CC_TAILBITING: 3>
CC_TERMINATED = <_cc_mode_t.CC_TERMINATED: 1>
CC_TRUNCATED = <_cc_mode_t.CC_TRUNCATED: 2>
const_lut = [2]
specinvert_lut = [[0, 2, 1, 3]]

FILE
/usr/local/lib/python3.12/dist-packages/gnuradio/fec/__init__.py

--- fft_help.txt ---

Help on package gnuradio.fft in gnuradio:

NAME
    gnuradio.fft - Fourier-transform blocks and related functions.

PACKAGE CONTENTS
    fft_python
    fft_vcc
    fft_vfc
    logpwrfft

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/fft/__init__.py

--- filter_help.txt ---

Help on package gnuradio.filter in gnuradio:

NAME
    gnuradio.filter - Filter blocks and related functions.

PACKAGE CONTENTS
    CustomViewBox
    GrFilterPlotWidget
    api_object
    bandgraphicsview
    banditems
    file_taps_loader
    filter_design
    filter_python
    filterbank
    fir_design
    freq_xlating_fft_filter
    icons_rc
    idealbanditems
    optfir
    pfb
    polezero_plot
    pyqt_filter_stacked

SUBMODULES
    kernel

FUNCTIONS
    pm_remez(...) method of builtins.PyCapsule instance
            pm_remez(order: int, bands: List[float], ampl: List[float], error_weight: List[float], filter_type: str = 'bandpass', grid_density: int = 16) -> List[float]

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/filter/__init__.py

--- iio_help.txt ---

Help on package gnuradio.iio in gnuradio:

NAME
    gnuradio.iio - Interface blocks for IIO devices

PACKAGE CONTENTS
    iio_python

FUNCTIONS
    get_pluto_uri(...) method of builtins.PyCapsule instance
        get_pluto_uri() -> str

DATA
    CHANNEL = <attr_type_t.CHANNEL: 0>
    DEVICE = <attr_type_t.DEVICE: 1>
    DEVICE_BUFFER = <attr_type_t.DEVICE_BUFFER: 2>
    DEVICE_DEBUG = <attr_type_t.DEVICE_DEBUG: 3>
    DIRECT_REGISTER_ACCESS = <attr_type_t.DIRECT_REGISTER_ACCESS: 4>
    DOUBLE = <data_type_t.DOUBLE: 0>
    FLOAT = <data_type_t.FLOAT: 1>
    INT = <data_type_t.INT: 3>
    LONGLONG = <data_type_t.LONGLONG: 2>
    UINT8 = <data_type_t.UINT8: 4>

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/iio/__init__.py

--- network_help.txt ---

Help on package gnuradio.network in gnuradio:

NAME
    gnuradio.network

DESCRIPTION
    This is the GNU Radio NETWORK module. Place your Python package
    description here (python/__init__.py).

PACKAGE CONTENTS
    network_python
    tcp_source

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/network/__init__.py

--- pdu_help.txt ---

Help on package gnuradio.pdu in gnuradio:

NAME
    gnuradio.pdu - Blocks and utilities for PDU based processing.

PACKAGE CONTENTS
    pdu_lambda
    pdu_python

DATA
    EARLY_BURST_APPEND = <early_pdu_behavior_t.EARLY_BURST_APPEND: 0>
    EARLY_BURST_BALK = <early_pdu_behavior_t.EARLY_BURST_BALK: 2>
    EARLY_BURST_DROP = <early_pdu_behavior_t.EARLY_BURST_DROP: 1>

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/pdu/__init__.py

--- qtgui_help.txt ---

Help on package gnuradio.qtgui in gnuradio:

NAME
    gnuradio.qtgui - Provides a GUI interface using the QT backend.

PACKAGE CONTENTS
    auto_correlator_sink
    azelplot
    compass
    dialcontrol
    dialgauge
    digitalnumbercontrol
    distanceradar
    graphicitem
    graphicoverlay
    ledindicator
    levelgauge
    msgcheckbox
    msgpushbutton
    qtgui_python
    range
    togglebutton
    toggleswitch
    util

DATA
    COMPLEX = <data_type_t.COMPLEX: 3>
    COMPLEX_VEC = <data_type_t.COMPLEX_VEC: 8>
    DOUBLE = <data_type_t.DOUBLE: 2>
    DOUBLE_VEC = <data_type_t.DOUBLE_VEC: 7>
    FLOAT = <data_type_t.FLOAT: 1>
    FLOAT_VEC = <data_type_t.FLOAT_VEC: 6>
    INT = <data_type_t.INT: 0>
    INTENSITY_COLOR_MAP_TYPE_BLACK_HOT = <intensity_t.INTENSITY_COLOR_MAP_...
    INTENSITY_COLOR_MAP_TYPE_COOL = <intensity_t.INTENSITY_COLOR_MAP_TYPE_...
    INTENSITY_COLOR_MAP_TYPE_INCANDESCENT = <intensity_t.INTENSITY_COLOR_M...
    INTENSITY_COLOR_MAP_TYPE_MULTI_COLOR = <intensity_t.INTENSITY_COLOR_MA...
    INTENSITY_COLOR_MAP_TYPE_SUNSET = <intensity_t.INTENSITY_COLOR_MAP_TYP...
    INTENSITY_COLOR_MAP_TYPE_USER_DEFINED = <intensity_t.INTENSITY_COLOR_M...
    INTENSITY_COLOR_MAP_TYPE_WHITE_HOT = <intensity_t.INTENSITY_COLOR_MAP_...
    INT_VEC = <data_type_t.INT_VEC: 5>
    NUM_GRAPH_HORIZ = <graph_t.NUM_GRAPH_HORIZ: 1>
    NUM_GRAPH_NONE = <graph_t.NUM_GRAPH_NONE: 0>
    NUM_GRAPH_VERT = <graph_t.NUM_GRAPH_VERT: 2>
    STRING = <data_type_t.STRING: 4>
    TRIG_MODE_AUTO = <trigger_mode.TRIG_MODE_AUTO: 1>
    TRIG_MODE_FREE = <trigger_mode.TRIG_MODE_FREE: 0>
    TRIG_MODE_NORM = <trigger_mode.TRIG_MODE_NORM: 2>
    TRIG_MODE_TAG = <trigger_mode.TRIG_MODE_TAG: 3>

TRIG_SLOPE_NEG = <trigger_slope.TRIG_SLOPE_NEG: 1>
TRIG_SLOPE_POS = <trigger_slope.TRIG_SLOPE_POS: 0>

FILE
/usr/local/lib/python3.12/dist-packages/gnuradio/qtgui/__init__.py

--- soapy_help.txt ---

Help on package gnuradio.soapy in gnuradio:

NAME
    gnuradio.soapy

DESCRIPTION
    This is the GNU Radio SOAPY module. Place your Python package
    description here (python/__init__.py).

PACKAGE CONTENTS
    soapy_python

DATA
    BOOL = <argtype_t.BOOL: 0>
    FLOAT = <argtype_t.FLOAT: 2>
    INT = <argtype_t.INT: 1>
    STRING = <argtype_t.STRING: 3>

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/soapy/__init__.py

--- trellis_help.txt ---

Help on package gnuradio.trellis in gnuradio:

NAME
    gnuradio.trellis - Blocks and utilities for trellis coding and related.

PACKAGE CONTENTS
    fsm_utils
    trellis_python

DATA
    TRELLIS_MIN_SUM = <siso_type_t.TRELLIS_MIN_SUM: 200>
    TRELLIS_SUM_PRODUCT = <siso_type_t.TRELLIS_SUM_PRODUCT: 201>

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/trellis/__init__.py

--- vocoder_help.txt ---

Help on package gnuradio.vocoder in gnuradio:

NAME
    gnuradio.vocoder

DESCRIPTION
    This is the gr-vocoder package. This package includes the various
    vocoder blocks in GNU Radio.

PACKAGE CONTENTS
    cvsd
    vocoder_python

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/vocoder/__init__.py

--- wavelet_help.txt ---

Help on package gnuradio.wavelet in gnuradio:

NAME
    gnuradio.wavelet - Processing blocks for wavelet transforms.

PACKAGE CONTENTS
    wavelet_python

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/wavelet/__init__.py

--- zeromq_help.txt ---

Help on package gnuradio.zeromq in gnuradio:

NAME
    gnuradio.zeromq - Blocks for interfacing with ZeroMQ endpoints.

PACKAGE CONTENTS
    probe_manager
    rpc_manager
    zeromq_python

FILE
    /usr/local/lib/python3.12/dist-packages/gnuradio/zeromq/__init__.py