

[Previous topic](#)[gnuradio.noaa](#)[Next topic](#)[gnuradio.qtgui](#)[This Page](#)[Show Source](#)[Quick search](#) Go

Enter search terms or a module, class or function name.

# gnuradio.pager

The GNU Radio pager application.

`gnuradio.pager.flex_deinterleave() → flex_deinterleave_sptr``flex deinterleave description``Constructor Specific Documentation:``Make flex deinterleaver.``flex_deinterleave_sptr.active_thread_priority(flex_deinterleave_sptr self)  
→ int``flex_deinterleave_sptr.declare_sample_delay(flex_deinterleave_sptr self, int which, int delay)  
declare_sample_delay(flex_deinterleave_sptr self, unsigned int delay)``flex_deinterleave_sptr.message_subscribers(flex_deinterleave_sptr self, swig_int_ptr which_port) → swig_int_ptr``flex_deinterleave_sptr.min_noutput_items(flex_deinterleave_sptr self) → int``flex_deinterleave_sptr.pc_input_buffers_full_avg(flex_deinterleave_sptr self, int which) → float  
pc_input_buffers_full_avg(flex_deinterleave_sptr self) -> pmt_vector_float``flex_deinterleave_sptr.pc_noutput_items_avg(flex_deinterleave_sptr self) → float``flex_deinterleave_sptr.pc_nproduced_avg(flex_deinterleave_sptr self) → float``flex_deinterleave_sptr.pc_output_buffers_full_avg(flex_deinterleave_sptr self, int which) → float  
pc_output_buffers_full_avg(flex_deinterleave_sptr self) -> pmt_vector_float``flex_deinterleave_sptr.pc_throughput_avg(flex_deinterleave_sptr self) → float``flex_deinterleave_sptr.pc_work_time_avg(flex_deinterleave_sptr self) → float``flex_deinterleave_sptr.pc_work_time_total(flex_deinterleave_sptr self) → float``flex_deinterleave_sptr.sample_delay(flex_deinterleave_sptr self, int which) → unsigned int``flex_deinterleave_sptr.set_min_noutput_items(flex_deinterleave_sptr self, int m)``flex_deinterleave_sptr.set_thread_priority(flex_deinterleave_sptr self, int priority) → int``flex_deinterleave_sptr.thread_priority(flex_deinterleave_sptr self) → int``gnuradio.pager.flex_frame() → flex_frame_sptr``flex_frame.``Constructor Specific Documentation:``gnuradio.pager.flex_parse(msg_queue_sptr queue, float freq) → flex_parse_sptr  
flex parse description`

Constructor Specific Documentation:

**Parameters:**

- **queue** –
- **freq** –

```
flex_parse_sptr.active_thread_priority(flex_parse_sptr self) → int  
flex_parse_sptr.declare_sample_delay(flex_parse_sptr self, int which, int delay)  
    declare_sample_delay(flex_parse_sptr self, unsigned int delay)  
flex_parse_sptr.message_subscribers(flex_parse_sptr self, swig_int_ptr which_port) → swig_int_ptr  
flex_parse_sptr.min_noutput_items(flex_parse_sptr self) → int  
flex_parse_sptr.pc_input_buffers_full_avg(flex_parse_sptr self, int which) → float  
    pc_input_buffers_full_avg(flex_parse_sptr self) -> pmt_vector_float  
flex_parse_sptr.pc_noutput_items_avg(flex_parse_sptr self) → float  
flex_parse_sptr.pc_nproduced_avg(flex_parse_sptr self) → float  
flex_parse_sptr.pc_output_buffers_full_avg(flex_parse_sptr self, int which) → float  
    pc_output_buffers_full_avg(flex_parse_sptr self) -> pmt_vector_float  
flex_parse_sptr.pc_throughput_avg(flex_parse_sptr self) → float  
flex_parse_sptr.pc_work_time_avg(flex_parse_sptr self) → float  
flex_parse_sptr.pc_work_time_total(flex_parse_sptr self) → float  
flex_parse_sptr.sample_delay(flex_parse_sptr self, int which) → unsigned int  
flex_parse_sptr.set_min_noutput_items(flex_parse_sptr self, int m)  
flex_parse_sptr.set_thread_priority(flex_parse_sptr self, int priority) → int  
flex_parse_sptr.thread_priority(flex_parse_sptr self) → int
```

gnuradio.page.**flex\_sync()** → flex\_sync\_sptr

flex sync description

Constructor Specific Documentation:

```
flex_sync_sptr.active_thread_priority(flex_sync_sptr self) → int  
flex_sync_sptr.declare_sample_delay(flex_sync_sptr self, int which, int delay)  
    declare_sample_delay(flex_sync_sptr self, unsigned int delay)  
flex_sync_sptr.message_subscribers(flex_sync_sptr self, swig_int_ptr which_port) → swig_int_ptr  
flex_sync_sptr.min_noutput_items(flex_sync_sptr self) → int  
flex_sync_sptr.pc_input_buffers_full_avg(flex_sync_sptr self, int which) → float  
    pc_input_buffers_full_avg(flex_sync_sptr self) -> pmt_vector_float  
flex_sync_sptr.pc_noutput_items_avg(flex_sync_sptr self) → float  
flex_sync_sptr.pc_nproduced_avg(flex_sync_sptr self) → float
```

```

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

flex_sync_sptr.pc_throughput_avg(flex_sync_sptr self) → float

flex_sync_sptr.pc_work_time_avg(flex_sync_sptr self) → float

flex_sync_sptr.pc_work_time_total(flex_sync_sptr self) → float

flex_sync_sptr.sample_delay(flex_sync_sptr self, int which) → unsigned int

flex_sync_sptr.set_min_noutput_items(flex_sync_sptr self, int m)

flex_sync_sptr.set_thread_priority(flex_sync_sptr self, int priority) → int

flex_sync_sptr.thread_priority(flex_sync_sptr self) → int

gnuradio.page.slicer_fb(float alpha) → slicer_fb_sptr
slicer description

Constructor Specific Documentation:

Make a pager slicer.

Parameters: alpha –
```

slicer\_fb\_sptr.**active\_thread\_priority**(slicer\_fb\_sptr self) → int

slicer\_fb\_sptr.**dc\_offset**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**declare\_sample\_delay**(slicer\_fb\_sptr self, int which, int delay)

declare\_sample\_delay(slicer\_fb\_sptr self, unsigned int delay)

slicer\_fb\_sptr.**message\_subscribers**(slicer\_fb\_sptr self, swig\_int\_ptr which\_port) → swig\_int\_ptr

slicer\_fb\_sptr.**min\_noutput\_items**(slicer\_fb\_sptr self) → int

slicer\_fb\_sptr.**pc\_input\_buffers\_full\_avg**(slicer\_fb\_sptr self, int which) → float
 pc\_input\_buffers\_full\_avg(slicer\_fb\_sptr self) -> pmt\_vector\_float

slicer\_fb\_sptr.**pc\_noutput\_items\_avg**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**pc\_nproduced\_avg**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**pc\_output\_buffers\_full\_avg**(slicer\_fb\_sptr self, int which) → float
 pc\_output\_buffers\_full\_avg(slicer\_fb\_sptr self) -> pmt\_vector\_float

slicer\_fb\_sptr.**pc\_throughput\_avg**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**pc\_work\_time\_avg**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**pc\_work\_time\_total**(slicer\_fb\_sptr self) → float

slicer\_fb\_sptr.**sample\_delay**(slicer\_fb\_sptr self, int which) → unsigned int

slicer\_fb\_sptr.**set\_min\_noutput\_items**(slicer\_fb\_sptr self, int m)

slicer\_fb\_sptr.**set\_thread\_priority**(slicer\_fb\_sptr self, int priority) → int

slicer\_fb\_sptr.**thread\_priority**(slicer\_fb\_sptr self) → int

