sig_filt = 'conv_valid_out' = convolve(b_firwin, sig_into_conv, mode='valid')

Dennis van Gils, 23-01-2019

Test scenario 1:

Perform convolve every incoming buffer, deque buffer is 2x BUFFER_SIZE

Not optimal

BUFFER_SIZE	10	samples	
N_buffers_in_deque	2	int	
N_deque	20	samples	
N_taps	7	samples	== must be odd number
N_sig_into_conv	16	samples	
offset_deque	4	samples	
N_conv_valid_out	10	samples	== BUFFER_SIZE
offset_valid	3	samples	
win_idx_valid_start	7	samples	

i_buffer:	#0: start-up										#1										#2										#3										
conv_valid_out:								1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3				
offset_valid:					1f	1f	1f								2f	2f	2f	1f	1f	1f					3f	3f	3f	2f	2f	2f								3f	3f	3f	
idx:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
					sig_into_conv			conv_valid_out						conv_valid_out				sig_into_conv																							
deque idx:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19																					
deque idx:											0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19											
deque idx:																					0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	

Test scenario 2:

Perform convolve every incoming buffer, deque buffer is 3x BUFFER_SIZE

Maximized N_taps given N_buffers_in_deque

BUFFER_SIZE	10	samples	
N_buffers_in_deque	3	int	
N_deque	30	samples	
N_taps	21	samples	== must be odd number
N_sig_into_conv	30	samples	
offset_deque	0	samples	
N_conv_valid_out	10	samples	== BUFFER_SIZE
offset_valid	10	samples	
win_idx_valid_start	10	samples	

i_buffer:	#0: start-up										#1: start-up										#2										#3										#4										
conv_valid_out:											2	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4											
offset_valid:	2f	2f	2f	2f	2f	2f	2f	2f	2f	2f	3f	3f	3f	3f	3f	3f	3f	3f	3f	3f	2f	2f	2f	2f	2f	2f	2f	2f	2f	2f	3f	3f	3f	3f	3f	3f	3f	3f	3f	3f	4f	4f	4f	4f	4f	4f	4f	4f	4f	4f	
																					4f	4f	4f	4f	4f	4f	4f	4f	4f	4f																					
idx:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
deque idx:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29																					
deque idx:											0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29											
deque idx:																					0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	

Optimal scenario:

Perform convolve every incoming buffer, maximize N_taps given N_buffers_in_deque

Fs	5000	Hz	
BUFFER_SIZE	500	samples	
N_buffers_in_deque	41	int	
N_deque	20500	samples	= BUFFER_SIZE * N_buffers_in_deque
N_taps	20001	samples	= BUFFER_SIZE * (N_buffers_in_deque - 1) + 1 == max that fits using N_deque

Fred Harris' approximation			Multirate Signal Processing for Communication Systems, Fredric J. Harris, 2004, page 216, equation (8.16)
f_pass	49	Hz	
f_stop	50	Hz	
filter attenuation	88.00	dB	= N_taps * 22 * ((f_stop - f_pass) / Fs)

N_sig_into_conv	20500	samples	= BUFFER_SIZE + N_taps - 1 == N_deque by optimal design	
offset_deque	0	samples	= N_deque - N_sig_into_conv == 0 by optimal design	
N_conv_valid_out	500	samples	= N_deque - N_taps + 1 == BUFFER_SIZE by optimal design	
offset_valid	10000	samples	= INT((N_taps - 1) / 2)	
win_idx_valid_start	10000	samples	= offset_valid + offset_deque == offset_valid by optimal design	
T_settle_filter	2.00	s	= win_idx_valid_start / Fs	i.e. Filter settling time
T_settle_deque	4.00	s	= T_settle_filter * 2	i.e. Deque settling time